OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
ANALYSIS/MODEL COVER SHEET
Complete Only Applicable Items

1. QA: QA
Page: 1 of: 52

2. ☒ Analysis — Check all that apply

Type of Analysis	☐ Engineering ☐ Performance Assessment ☒ Scientific
Intended Use of Analysis	☐ Input to Calculation ☒ Input to another Analysis or Model ☒ Input to Technical Document

Describe use:

Provides comparison of transport solutions for uncertainties in results for an AMR and the UZ PMR.

3. ☐ Model — Check all that apply

Type of Model	☐ Conceptual Model ☐ Abstraction Model ☐ Mathematical Model ☐ System Model ☐ Process Model
Intended Use of Model	☐ Input to Calculation ☐ Input to another Model or Analysis ☐ Input to Technical Document

Describe use:

4. Title:

Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking

5. Document Identifier (including Rev. No. and Change No., if applicable):

ANL-NBS-HS-000001 Rev 00

6. Total Attachments:

3

7. Attachment Numbers - No. of Pages in Each:

I-11, II-5, III-91

	Printed Name	Signature	Date
8. Originator	Lehua Pan Cliff Ho	Lehua Pan Mal Ng for Cliff Ho	3/9/00 3/9/00
9. Checker	Peter Persoff	Peter Persoff	3/9/00
10. Lead/Supervisor	G. S. Bodvarsson	B0 R	3/9/00
11. Responsible Manager	G. S. Bodvarsson	B0 R	3/9/00

12. Remarks:

Initial Issue

Block 8: this AMR was prepared by Lehua Pan except for section 6.1 and parts of section 3 and section 6.4.3 relating to FEHM, which were prepared by Cliff Ho. Simulations using TOUGH2 V1.4 and T2R3D V1.4 were performed by Y. S. Wu.

Editorial correction to Attachment I – added "N/A" to "Change" box throughout LEH 4/5/00
Editorial correction to Attachment III – pagination throughout LEH 4/5/00

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
ANALYSIS/MODEL REVISION RECORD

Complete Only Applicable Items

2. Analysis or Model Title:

Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking

3. Document Identifier (including Rev. No. and Change No., if applicable):

ANL-NBS-HS-000001 Rev 00

4. Revision/Change No.	5. Description of Revision/Change
00	Initial Issue

CONTENTS

Page

ACRONYMS 9

1. PURPOSE 11

2. QUALITY ASSURANCE 13

3. COMPUTER SOFTWARE AND MODEL USAGE 15

4. INPUTS 17
   4.1 DATA AND PARAMETERS 17
   4.2 CRITERIA 17
   4.3 CODES AND STANDARDS 18

5. ASSUMPTIONS 19
   5.1 INPUT DATA 19
   5.2 TRANSPORT PROCESSES 19
   5.3 DISCRETIZATIONS 19

6. ANALYSIS/MODEL 21
   6.1 DESCRIPTION OF PARTICLE-TRACKING IN FEHM 22
   6.2 DESCRIPTION OF PARTICLE-TRACKER DCPT 23
   6.3 DESCRIPTION OF ADVECTIVE-DISPERSIVE SOLUTIONS WITH T2R3D 29
   6.4 COMPARISONS OF FEHM AND DCPT WITH T2R3D 31

7. CONCLUSIONS 45

8. INPUTS AND REFERENCES 47
   8.1 DOCUMENTS CITED 47
   SOFTWARE CITED 48
   8.2 STANDARDS, CODES, REGULATIONS AND PROCEDURES CITED 48
   8.3 SOURCE DATA, LISTED BY DATA TRACKING NUMBER 49
   8.4 AMR OUTPUT DATA LISTED BY DATA TRACKING NUMBER 49

9. ATTACHMENTS 51
   ATTACHMENT I - DOCUMENT INPUT REFERENCE SHEET I-1
   ATTACHMENT II - INPUT AND OUTPUT FILES FOR DCPT AND FEHM II-1
   1. FILES FOR DCPT II-1
   2. FILES FOR FEHM II-5
   ATTACHMENT III - SOFTWARE ROUTINES III-1

INTENTIONALLY LEFT BLANK

FIGURES

Page

1. Comparison between DCPT and the Analytical Solution for a Parallel Fracture System 33

2. Comparison between DCPT and the Analytical Solution for a 2-D Transport Problem with Dispersion Tensors. 35

3. Comparison between DCPT and T2R3D for 1-D Radionuclide Transport. (a) Technetium, (b) Neptunium 37

4. Normalized Cumulative Breakthrough of Technetium at the Water Table for FEHM and T2R3D.....39

5. Normalized Cumulative Breakthrough of Neptunium at the Water Table for FEHM and T2R3D 39

6. Map View of the 3-D Grid 42

7. Comparison between DCPT and T2R3D for 3-D Radionuclide Transport of Technetium 43

8. Comparison between DCPT and T2R3D for 3-D Radionuclide Transport of Neptunium 43

INTENTIONALLY LEFT BLANK

TABLES

Page

1. Table of Software Used in This Analysis 15

2. Input Data 17

3. Scientific Notebooks 22

4. Parameters Used in Transport Problem in a Parallel Fracture System 32

5. Parameters of the 2-D Case 34

6. Parameters Used for 1-D Radionuclide Transport 36

7. Four Transport Simulations Used in FEHM vs. T2R3D Comparison 38

INTENTIONALLY LEFT BLANK

ACRONYMS

1-D	One-Dimensional
2-D	Two-Dimensional
3-D	Three-Dimensional
ACC	Accession Number
A-D	Advective-Dispersive
AMR	Analysis/Model Report
AP	Administrative Procedure (DOE)
CHn	Calico Hills nonwelded unit
CRWMS	Civilian Radioactive Waste Management System
DCPT	Dual Continuum Particle Tracker
DIRS	Document Input Reference Sheet
DKM	Dual-Permeability Model
DOE	Department of Energy
DTN	Data Tracking Number
FEHM	Finite Element Heat and Mass
FY	Fiscal Year
LA	License Application
LBNL	Lawrence Berkeley National Laboratory
M&O	Management and Operating Contractor
OCRWM	Office of Civilian Radioactive Waste Management
PA	Performance Assessment
PMR	Process Model Report
PTn	Paintbrush nonwelded unit
QAP	Quality Administrative Procedure (M&O)
QARD	Quality Assurance Requirements and Description
QIP	Quality Implementing Procedure
RIS	Records Information System
SR	Site Recommendation
STN	Software Tracking Number
TBV	To Be Verified
TCw	Tiva Canyon welded unit
TDMS	Technical Data Management System
TSw	Topopah Spring welded unit
TSPA	Total System Performance Assessment

ACRONYMS (Continued)

UZ	Unsaturated Zone
VA	Viability Assessment
YMP	Yucca Mountain Site Characterization Project

1. PURPOSE

The purpose of this Analysis/Model Report (AMR) is to compare transport simulations utilizing particle-tracking methods with simulations using the more rigorous fully coupled advective-dispersive (A-D) approach. This is in accordance with *AMR Development Plan for U0155 Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking* (CRWMS 1999a). The fully coupled A-D flow and transport simulations incorporate advection, dispersion, sorption, and decay processes. These are compared with results from particle-tracking methods including the method used for the Total System Performance Assessment (TSPA) for the Viability Assessment (VA). This AMR supports the Unsaturated Zone (UZ) Flow and Transport Process Model Report (PMR) as well as other AMRs.

In this AMR, two particle-tracking methods are compared with the A-D approach. The results of (1) the Finite Element Heat and Mass (FEHM) particle-tracking code (FEHM, Software Tracking Number (STN): 10031-2.00-00, Version 2.0), which was used for TSPA-VA, and (2) the random-walk particle-tracking code, Dual Continuum Particle Tracker (DCPT, STN: 10078-1.0-00, Version 1.0), are compared to the results from the code T2R3D (T2R3D, STN: 10006-1.4-00, Version 1.4), a fully coupled A-D numerical code.

The constraints and limitations of the results presented here are that the radionuclide breakthrough curves presented should not be considered to be predictions of radionuclide transport in the UZ at Yucca Mountain. The results are for comparison purposes only and the input values used in the comparisons are not necessarily the same as those that will be used in TSPA for Site Recommendation (SR) and License Application (LA). The analysis and simulations, though, do utilize inputs representative of the range of conditions at Yucca Mountain, but these are not necessarily the final properties to be used in the UZ PMR and TSPA-SR/LA. Predictions for the radionuclide breakthrough curve for the UZ for TSPA-SR/LA will be provided in future AMRs and the UZ PMR. It should also be noted that because the effect of radioactive decay would be essentially the same for all of the methods being compared here, it was not necessary to include this process in the comparisons presented here.

INTENTIONALLY LEFT BLANK

2. QUALITY ASSURANCE

This AMR was developed in accordance with AP-3.10Q, *Analyses and Models*. Other applicable DOE Administrative Procedures (APs) and YMP-LBNL Quality Implementing Procedures (QIPs) are identified in *AMR Development Plan for U0155 Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking* (CRWMS M&O 1999a).

This analysis was evaluated with other related activities in accordance with QAP-2-0, *Conduct of Activities*, and determined to be quality-affecting and subject to the requirements of the QARD*, Quality Assurance Requirements and Description* (DOE 1998). This evaluation is documented in *Activity Evaluation of M&O Site Investigations* (CRWMS M&O 1999b,c). The activity evaluation (per QAP-2-0) completed for performance-assessment activities was also determined to be quality affecting and is documented in *Conduct of Performance Assessment* (CRWMS M&O 1999d).

INTENTIONALLY LEFT BLANK

3. COMPUTER SOFTWARE AND MODEL USAGE

The software codes and routines used in this study are listed in Table 1. These are appropriate for the intended application and were used only within their range of software validation in accordance with AP-SI.1Q, Rev. 1, ICN 0, *Software Management*. The DCPT (DCPT, STN: 10078-1.0-00, Version 1.0) and FEHM (FEHM, STN: 10031-2.00-00, Version 2.0) codes are used to simulate transport of radionuclides using particle-tracking techniques. T2R3D (T2R3D, STN: 10006-1.4-00, Version 1.4) is used to perform numerical simulations for comparison to the particle-tracking code results. The software code TOUGH2 (TOUGH2, STN: 10007-1.4-00, Version 1.4) is used to generate flow fields for input to the transport codes. The Q-status of these codes and macros is listed in Attachment I and discussed below.

Table 1. Table of Software Used in This Analysis

Software Name	Version	Software Tracking Number (STN)	Computer Type
FEHM	2.0	10031-2.00-00	UNIX
DCPT	1.0	10078-1.0-00	PC w/Windows 95
T2R3D	1.4	10006-1.4-00	Sun Workstation w/UNIX
TOUGH2	1.4	10007-1.4-01	Sun Workstation w/UNIX
Routines:		**ACC**:	
T2FEHM2	2.0	MOL. 19990915.0359	UNIX
PROCESS1	1.0	MOL. 19990915.0360	UNIX
MAKEPTRK	1.0	MOL. 19990915.0361	UNIX
PrepareKDfile	1.0	MOL. 20000127.0120	PC
ExtractFlow	1.0	MOL. 20000127.0121	PC
ExBT	1.0	MOL. 20000127.0122	PC
StatSpatial	1.0	MOL. 20000202.0193	PC

TOUGH2 (Version 1.4) and T2R3D (Version 1.4) have been qualified under AP-SI.1Q and were obtained from configuration management. The use of TOUGH2 (Version 1.4) and T2R3D (Version 1.4) prior to obtaining them from configuration management is being evaluated under AP-3.17Q, *Impact Reviews*, but no impact is anticipated. FEHM (Version 2.0) was qualified prior to the effective date of AP-SI.1Q. It has been reverified and was obtained from configuration management per AP-SI.1Q. DCPT (Version 1.0) is being qualified and a Software Activity Plan for use of unqualified software and copy of the code have been submitted to configuration management per Section 5.12 of AP-SI.1Q, Rev. 2, ICN 2.

T2FEHM2 (Version 2.0), PROCESS1 (Version 1.0), and MAKEPTRK (Version 1.0) are single-user software routines qualified per AP-SI.1Q, Rev. 1, ICN 0 and the documentation has been submitted to the Records Processing Center (RPC), the TDMS and is included in Attachement III. PrepareKDfile (Version1.0), ExtractFlow (Version 1.0), ExBT (Version 1.0) and StatSpatial (Version 1.0) were qualified per AP-SI.1Q and the documentation has been submitted to the RPC and is included in Attachment III. T2FEHM2 is a routine written to create FEHM-readable files from TOUGH2 output flow fields. PROCESS1 is a software routine that post-processes the results of the FEHM particle-tracking simulation to provide columns of time versus mass flux and cumulative mass at the water table. MAKEPTRK creates a transport parameter data file for FEHM to read in the particle-tracking simulation. PrepareKDfile is a routine written to create a DCPT-readable file from a TOUGH2 mesh file and T2R3D input file. ExtractFlow is a routine written to create a DCPT-readable file from a TOUGH2 output file. ExBT is a routine written to extract a breakthrough curve from the T2R3D output file. StatSpatial is used to calculate the distribution of particles along a user-specified line based on the DCPT output file. Grids from the UZ Flow and Transport Model are used for comparing these transport codes.

Input and output files for this AMR are provided in Attachment II.

The commercially-available graphics plotting program Tecplot (Version 7.0) and the plotting portion of KaleidaGraph v.3.09 were also used but are not subject to software qualification assurance requirements.

4. INPUTS

4.1 DATA AND PARAMETERS

The input data used in this AMR are summarized in Table 2. The Q-status of these data is provided in Attachment I.

Table 2. Input Data

DTN	Description
LB971212001254.001	DKM Basecase Parameter Set for UZ Model, FY97 (Used for FEHM and TOUGH2 Input Parameters)
LB997141233129.001	Calibrated Basecase Infiltration 1-D Parameter Set for the UZ Model, FY99. (Used for TOUGH2/DCPT and T2R3D Input Parameters)
LB990501233129.004	3-D grid (FY99) used for T2R3D

The transport simulations comparing T2R3D and the FEHM particle-tracking method use the hydrologic base-case parameter set (DTN: LB971212001254.001) that was used for TSPA-VA. The values used for the sorption coefficients, diffusion coefficients, and dispersivities are given in Section 6.4.3. The precise values of these flow and transport parameters are not considered inputs that require additional verification because the purpose of this analysis is not to document specific transport simulation results, but to compare several transport simulation methodologies for the same transport system.

The one-dimensional (1-D) computational grid representing borehole USW SD-9, used in transport simulations comparing the DCPT and FEHM particle-tracking methods to T2R3D results, was obtained from the grid used for TSPA-VA and was used for comparison purposes only. The extraction of this 1-D column is documented in the Scientific Notebook YMP-LBNL-GSB-1.6.3 (pp. 39-40).

All input files are listed in Attachment II (DTN: LB990901233129.001 & DTN: SN9908T0581699.001).

4.2 CRITERIA

At this time, no specific criteria (e.g., System Description Documents) have been identified as applying to this analysis in project requirements documents. However, this AMR provides information required in specific subparts of the proposed U.S. Nuclear Regulatory Commission rule 10 CFR 63 (see Federal Register for February 22, 1999, 64 FR 8640). It supports the technical basis for methodologies used in performance assessment by comparing outputs with other detailed process-level methodologies (Subpart E, Section 114).

The DOE interim guidance (Dyer 1999), requiring the use of specified subparts of the proposed NRC high-level waste rule, 10 CFR Part 63 (64 FR 8640), was released after completion of the work documented in this AMR; it has no impact on this work activity.

4.3 CODES AND STANDARDS

No specific formally established standards have been identified as applying to this analysis.

5. ASSUMPTIONS

This AMR evaluates three numerical simulators by comparing their outputs for radionuclide transport problems, using the input data given in Section 4. The results of these simulations are not to be considered as predictions of transport from a potential nuclear waste repository because the input data are not necessarily the final input values that will be used for TSPA-SR/LA, and because radioactive decay is not included in these simulations. Radioactive decay is handled exactly the same by all simulators and has been ignored because this simplifies the comparisons between the simulation outputs.

Any numerical simulator is a simplification or approximation of the physical world. This section lists the principal simplifications and approximations that are used by all the simulators tested in this AMR. It is assumed that these simplifications do not significantly distort the outputs.

5.1 INPUT DATA

It is assumed that the input data are sufficiently representative of the conditions at Yucca Mountain that the comparison among the simulators and the findings of this AMR would not change if the input data used for TSPA-SR/LA were not identical to those used here. This assumption is based on several years of evaluations by many investigators and considered to be the only available source of the data. This assumption is used throughout this AMR and requires no further justification.

5.2 TRANSPORT PROCESSES

The transport processes included in this analysis are those that were used in TSPA-VA, except for radioactive decay. These are: advection, diffusion and dispersion, and equilibrium sorption of solutes. Radioactive decay has been ignored to facilitate comparisons between the simulation outputs. It is assumed that inclusion of radioactive decay would not significantly affect the comparison among the methods. This assumption is justified because radioactive decay is mathematically simple and is handled identically by all the simulators. This assumption is used throughout Section 6 and requires no further justification.

5.3 DISCRETIZATIONS

All standard numerical flow and transport simulators, including those used here, rely upon spatial and temporal discretization, and therefore provide spatially and temporal approximations of the natural system (Wu et al. 1999, pp. 190-193). Also, the methods tested here use dual-permeability grids, described in Section 6 (Wu et al. 1999, pp. 187-188, Doughty 1999, pp. 100-104). It is assumed that the spatial and temporal discretizations, and the appropriate use of dual-permeability grids, do not cause significant errors and do not distort the comparisons among the methods. This assumption is justified by the process of grid development, in which various degrees of grid refinement are tested until further refinement yields little improvement. This assumption requires no further justification.

INTENTIONALLY LEFT BLANK

6. ANALYSIS/MODEL

Transport calculations are integral parts to the simulation and prediction of the movement of radionuclides in the UZ. The UZ Model is formulated to rigorously solve both the transport conservation equations and the flow equations using finite-difference techniques. However, as the complexity of the model increases, solving the full transport equations becomes computationally intensive. An alternative approach that is generally less computationally intensive is the use of a particle-tracking method. In addition, compared with finite-element or finite-difference methods, particle-tracking methods usually give better spatial resolution, eliminate numerical dispersion effects, and reduce large truncation errors. However, particle-tracking approaches can vary according to the methods for describing the movement of particles and the assumptions used to determine their interaction with the flow field. Particularly, the exchange of mass between the fractures and matrix in the UZ makes the implementation of particle-tracking approaches more complicated. Therefore, it must be demonstrated that the particle-tracking approach yields acceptable results relative to the more rigorous fully coupled advective-dispersive transport approach.

For this AMR, transport simulations are performed with two particle-tracking methods. One is the residence-time-transfer function particle-tracking method of Finite Element Heat and Mass (FEHM, STN: 10031-2.00-00, Version 2.0) that was utilized in the TSPA-VA. The other is the random-walk particle-tracking method used in the Dual Continuum Particle-Tracker (DCPT, STN: 10078-1.0-00, Version 1.0). The FEHM particle-tracking method has been described in the FEHM User's Manual (FEHM, STN: 10031-2.00-00, Version 2.0) while DCPT is described in this AMR as well as in its software qualification package (DCPT, STN: 10078-1.0-00, Version 1.0). Transport simulations are performed to compare the DCPT to transport problems with analytical solutions and advective-dispersive numerical results using T2R3D (T2R3D, STN: 10006-1.4-00, Version 1.4). Other transport simulations are performed to compare the results using the FEHM particle-tracking method to T2R3D results for a 1-D column. The cumulative breakthrough curves of two radionuclides (one sorbing and one nonsorbing) are compared using the different methods. All test cases used for comparisons with T2R3D simulations use the realistic Yucca Mountain geology from the UZ Model. The results are evaluated for differences between the three approaches, and assessments of the impacts of the differences are provided. Radioactive decay is not included in this comparison analysis because the effect of radioactive decay would be essentially the same for all of the methods being compared here.

To facilitate simulation of water flow and solute transport in the fractured porous media, dual-permeability grids are used for all methods in this AMR. In a dual-permeability grid, the problem domain is represented by two overlapping grids, respectively representing the matrix continuum and the fracture continuum. Water or solute can flow between adjacent grid cells in one grid (in the same continuum) or between the two grid cells in different grids that overlap each other (between two continua). This mass transfer between fracture and matrix is a unique feature of transport in fractured porous media. Because the pore-water velocities in the fracture and matrix continua can differ by orders of magnitude, correct simulation of mass transfer between the two continua is one of the key factors that determine the success of a numerical model. In this AMR, the same dual-permeability grid is used for each case, but the approaches used to model the mass

transfer between the fracture continuum and the matrix continuum differ among the three methods. The detailed descriptions are provided in relevant sections (Sections 6.1, 6.2, and 6.3).

Key scientific notebooks (with relevant page numbers) used for the analysis described in this AMR are listed in Table 3.

Table 3. Scientific Notebooks

LBNL Scientific Notebook ID	M&O Scientific Notebook ID	Page Numbers
YMP-LBNL-GSB-LHH-1	SN-LBNL-SCI-035-VI	83 – 89
YMP-LBNL-GSB-LP-3	SN-LBNL-SCI-155-VI	1 – 105
YMP-LBNL-YSW-2	SN-LBNL-SCI-120-VI	106-108
YMP-LBNL-GSB-1.6.3	SN-LBNL-SCI-085-VI	39-40

6.1 DESCRIPTION OF PARTICLE-TRACKING IN FEHM

A complete description of the FEHM particle-tracking model can be found in the Models and Methods Summary for the FEHM software qualification package (FEHM, STN: 10031-2.00-00, Version 2.0) and in AMR U0065 (CRWMS M&O 2000b). Only a brief summary from those documents is provided here.

The particle-tracking method in FEHM views the computational domain as an interconnected network of fluid storage volumes. The two steps in the particle-tracking approach for steady-state flow fields are: (1) determine the time a particle spends in a cell, and (2) determine which cell the particle travels to next. The domain can consist of a single-continuum or dual-continua (e.g., fracture plus matrix) representation of the flow field.

The time that a particle spends in a cell is a function of the mass of liquid in that cell, the mass flow rates out of that cell into neighboring cells, and the diffusive, dispersive, and sorptive processes within that cell. For advective flow only, the residence time is uniquely defined by the ratio of the mass of liquid in a cell to the sum of the mass flow rates out of that cell. However, dispersive, diffusive, and sorptive processes provide distributions of particle “breakthrough” times for each cell, which are used to determine the effective residence time for a particle in each cell. The standard advection-dispersion equation (with sorption) is used to evaluate the breakthrough times for each cell. If diffusion into an adjacent matrix cell occurs, a one-dimensional diffusion equation for transport between the fracture cell and the matrix cell is also included. The analytic solution for diffusion into the matrix cell in the current particle-tracking model assumes an infinite domain.

The analytic solutions for the advection-dispersion equation with possible diffusion into a matrix cell yield cumulative, normalized breakthrough concentrations for each cell as a function of time. These curves also represent the cumulative distribution functions for the residence time of a particle that experiences advection, dispersion, diffusion, and sorption in each cell. A random number generator is then used to select a value between 0 and 1, which prescribes a particle

residence time from the cumulative distribution functions. The cumulative distribution function for the residence times is accurately represented with a sufficiently large number of particles that pass through the cell.

The probability of a particle traveling to a neighboring cell is proportional to the advective mass flow rate to each neighboring cell. Only outflows from a cell are considered; therefore, the probability of traveling to a cell that has mass flow coming into the current cell is zero. The mass flow rate to an adjacent cell divided by the total mass flow rate out of the current cell is equal to the probability that the particle will travel to that cell. A cumulative distribution function is derived from all the probabilities, and a random number selected between 0 and 1 therefore prescribes the cell to which a particle will travel. Again, a sufficiently large number of particles are used to reproduce the appropriate cumulative distribution function.

As described above, the FEHM particle-tracker simulates the advective portion and the diffusive portion of the fracture-matrix mass transfer separately. The advective portion of mass flow between the fracture and the matrix is accounted for by calculating the probability of a particle traveling to a neighboring cell (the matrix cell is treated as one of the neighboring cells to the fracture cell, vise verse). Therefore, the probability of a particle traveling from a fracture cell to a matrix cell is proportional to the advective mass flow rate in the same direction. However, the FEHM particle-tracking algorithm yields only additional residence time (a retardation) for the particles in the fracture that experience diffusive mass flow from fracture into matrix, but the particles do not actually get transported into the matrix. The additional residence time is calculated based on an analytical solution for a single fracture system (Tang et al. 1981, pp. 555-564). This model implies that the particles diffusing into the matrix cannot move vertically unless they first diffuse back to the fractures.

Though FEHM particle tracker has this capability, radioactive decay is not used in this comparison because the effect of radioactive decay would be essentially the same for all the methods being compared here.

6.2 DESCRIPTION OF PARTICLE-TRACKER DCPT

6.2.1 General Approaches and Overall Structures

The random-walk particle tracker DCPT describes the history of individual particles instead of focusing on fixed points of space. It uses the Lagrangian point of view, not the Eulerian point of view. The movement of a plume is described as a sum of the movements of individual particles. The coordinates of a moving particle are represented as functions of time (Bear 1972, p. 70, Equation 4.1.18):

$$\mathbf{X} = \mathbf{X}(\xi, t) \qquad \text{(Eq. 1)}$$

where $\mathbf{X}$ and ξ are the vectors that describe the positions of the particle at time t and some initial time (e.g., $t = 0$), respectively. Note that $\mathbf{X}$ is the dependent variable (vector) in Equation 1 while the function includes factors such as velocity, dispersion coefficient, and adsorption parameters.

The velocity, dispersion coefficients, and adsorption parameters are generally functions of space. These data are provided as tables of values on a discretized space (e.g., a grid). DCPT transforms these fixed-space values in the Eulerian point of view into the parameters of Equation (1) in the moving-particle (Lagrangian) point of view. Because the whole domain is discretized into subdomains or grid cells, the velocity field, or other fields of parameters, can also be disassembled in the same way. Cells are the basic units of a domain. Each cell has two sets of parameters, each of them corresponding to one continuum, and one set of parameters that defines the interactions between two continua. In dual-continua media (i.e., fractured-porous rock), a particle will travel either in the fracture continuum or in the matrix continuum, two overlapping continua often with very different velocities and parameters. The random switch between the fracture and the matrix is governed by a particle-transfer probability that should be consistent with the mass flow between two continua within that cell.

The object-oriented-program approach is used in developing the DCPT. Two major objects are used in DCPT. One is called CELL, which has all the information of the continua (e.g., the geometry, local velocities, dispersion coefficient tensor, and other parameters for both fracture and matrix). The other is called PARTICLE, which has properties describing the current status of a particle including the current time, the current XYZ position, the current cell, and the current continuum (fracture or matrix). Therefore, the major algorithm of particle tracking for a given particle and a given time step can be summarized as below:

1. Calculate the displacement that the particle will take during the time step based on the current status of the particle (see Section 6.2.2);

2. Determine whether the path of the particle intersects with any face of the current cell; if it does not, go to Step (3); otherwise, use the intersection point as the new location of the particle, reduce the time step accordingly, and get the neighboring cell ID;

3. Determine whether the particle will switch to the other continuum at the next time step (see Section 6.2.4);

4. Update the status of the particle with the results of Steps 2 and 3;

5. Check whether the particle has exited through the domain boundary or the specified maximum time has been passed; if yes, finish the simulation of this particle, otherwise go to Step 1.

In short, DCPT simulates the random walk of particles in a continuous space with discretized continua (cell based), but uses the particle-transfer probability to control which continuum a particle will travel in at a particular time. The following are some details of the approaches used in DCPT.

6.2.2 Calculation of Particle Displacement

The new location of a particle after a time step Δt is a random vector and can be calculated as (LaBolle et al. 1996, Equation 3, p. 584, symbolically replacing X_p and Δw with X and $W\sqrt{\Delta t}$, respectively)

$$X(t+\Delta t) = X(t) + A\Delta t + BW\sqrt{\Delta t} \qquad \text{(Eq. 2)}$$

The drift term A (see LaBolle et al. 1996, Equation 10, p. 584) is approximated to be the local pore velocity V. The tensor B and its transpose B^T are given by BB^T = 2D where D is the local dispersion coefficient tensor. W is a random vector, each component of which observes the $N(0,1)$ distribution. For simplicity, two additional terms in drift term A related to the divergence of D and the gradient of the volumetric water content are neglected. As shown in Sections 6.4.3 and 6.4.4, this approximation is acceptable for the advection-dominant transport processes in a steady-state flow field, such as was used for TSPA-VA. For a particle, the mean displacement vector is $V\Delta t$ while the variance tensor is $2D\Delta t$ in a given continuum. Whether the properties of the fracture or those of the matrix are used in Equation 2 depends on which continuum the particle travels in.

6.2.3 Sorption and Decay

For a reactive solute, only a portion of particles are mobile as described by Equation 2 with the remainder being sorbed. The probability, P_r, of a particle being in fluid can be defined as:

$$P_r = \frac{\theta}{\theta + (1-\phi) K_d \rho_R} \qquad \text{(Eq. 3)}$$

where K_d , ϕ, and ρ_R are the sorption distribution coefficient (m^3/kg), the porosity (m^3/m^3) and the rock density (kg/m^3) of the particular continuum, respectively; and θ is the volumetric water content. In terms of implementing the sorption process in particle tracking, we can take P_r in a deterministic way, as the percentage of the total mass of a moving particle. Therefore, the effective displacement of the particle will be P_r times the original displacement, which can be implemented by simply multiplying A and B in Equation 2 by P_r .

To simulate the radioactive decay, the mass of each particle, M_p, is calculated as a function of time, t:

$$M_p(t) = M_p(0) 2^{-t/t_{0.5}} \qquad \text{(Eq. 4)}$$

where $t_{0.5}$ is the half-life. Though DCPT has this capability, radioactive decay is not used in this comparison because the effect of radioactive decay would be essentially the same for all of the methods being compared here.

6.2.4. Particle-Transfer Probability: Mass Transfer between Fracture and Matrix

The mass-transfer process between fractures and matrix is simulated by random particle exchanges between two continua as controlled by the particle-transfer probabilities of either fracture-to-matrix or matrix-to-fracture progression, as described in Step 3 in Section 6.2.1. For other variables such as velocity and dispersion coefficients, grid cells are used in DCPT as the basic units for evaluating the particle-transfer probability. For each net mass flow between two continua in the fixed-space Eulerian point of view, there are two corresponding particle-transfer probabilities in the moving-particle Lagrangian point of view. One is the particle-transfer probability of particles from fracture to matrix, and the other is that from matrix to fracture. The challenge is how to transform correctly the net mass flow in the Eulerian point of view into two separate particle-transfer probabilities in the Lagrangian point of view. In the following derivation, we focus on the particle-transfer probability P_{fm} from a fracture to the matrix. The other probability P_{mf} can be similarly derived.

If the particles in the fracture continuum of a given grid cell at t = 0 have mass M_0, and the fraction of them that enter into the matrix continuum during the time interval (0, t) have mass M_{fm}, the particle-transfer probability P_{fm} can be defined as:

$$P_{fm} = \frac{M_{fm}}{M_0} \qquad \text{(Eq. 5)}$$

For a single particle in the fracture at t = 0, P_{fm} is the probability at which it will be in the matrix at time t. M_{fm} is directly proportional to the mass flow from fracture to matrix.

For a given grid cell, the net solute mass J transferred from fracture to matrix during a small time interval dt through a small area of the interface dA is:

$$d\,J_{fm} = \left[\max(q_{fm} C_f, 0) - \max(-q_{fm} C_m, 0) - D \frac{\partial C_m}{\partial s}\Big|_{s=0} \right] dA\, dt \qquad \text{(Eq. 6)}$$

where q_{fm} is the water flux (L/T) between fracture and matrix and $\vec{n}$ is the normal vector of the interface and points from fracture to matrix. This being the case, only one of the two advection terms in Equation 6 will take effect, depending on the direction of water flow. C is concentration and D is the dispersion coefficient specifically for the fracture-matrix interaction. A and t are fracture-matrix interfacial area and time, respectively. The variable s is the distance away from the fracture-matrix interface (s = 0 at the interface). Because in reality the detailed geometry of the interface and those variables defined on the interface are not available, it is not practical to derive a formulation to calculate the total mass flow between fracture and matrix (even in cell-scale)

without some simplifications. In DCPT, a lumping approach similar to that in T2R3D is used to estimate the net mass transfer between the fracture and the matrix at the grid-cell scale. Assuming (as in Section 5.3) that all dependent variables and the parameter D can be used in a sense of average values within the grid cell or over the interface, we can get the net mass transfer during the time interval (0, t) by integration of Equation 6 over the whole interface area:

$$J_{fm} = \int_0^t [\ \max(Q_{fm}\ C_f\ , 0) - \max(-Q_{fm}\ C_m\ , 0) + \frac{D}{S}(\ C_f - C_m)\ A]\ d\tau \qquad \text{(Eq. 7)}$$

where S is the characteristic distance of the fracture-matrix system proportional to the fracture spacing (e.g., 1/6 of fracture spacing depending on the assumptions of the fracture network). Q_{fm} is the net water flow rate (M/T) between fracture and matrix. Its value is positive if the mass flows from fracture to matrix. Note that t is a particular time, i.e., the end of a time step, while τ is the variable of integration.

Equation 7 can be rewritten as:

$$\begin{aligned} J_{fm} &= \int_0^t [\ \max(q_{fm}\ , 0) + \frac{D}{S}\ ]\ A\ C_f\ \ d\tau - \int_0^t [\max(-q_{fm}\ , 0) + \frac{D}{S}\ ]\ A\ C_m\ d\tau \\ &= F_{fm} \int_0^t C_f\ \ d\tau + F_{mf} \int_0^t C_m\ \ \ d\tau \end{aligned} \qquad \text{(Eq. 8)}$$

F_{fm} is the effective flow rate from the fracture to the matrix while F_{mf} is the effective flow rate from the matrix to the fracture.

Equation 8 states that the net mass transfer from fracture to matrix is the total mass flow from fracture to matrix less the total mass flow from matrix to fracture.

The first term on the right hand of Equation 8 is the mass flow from fracture to matrix during the time interval (0, t). However, it is not M_{fm} in Equation 5 because C_f includes not only the particles that are in the fracture at t = 0, but also those particles that enter the fracture during (0, t) from either the matrix or other neighboring blocks. In other words, if C^E is the concentration of the particles that entered the continuum during (0, t) and C^{NE} is the concentration of the particles already in the continuum at t = 0, we can split the first term on the right hand of Equation 8 as:

$$F_{fm} \int_0^t C_f\ \ d\tau = F_{fm} \int_0^t {C_f}^E\ d\tau + F_{fm} \int_0^t {C_f}^{NE}\ d\tau \qquad \text{(Eq. 9)}$$

Equation 9 simply states that only a portion of the mass flow from the fracture to the matrix consists of the particles that were initially in the fracture (the second term). Only this portion is needed to calculate the probability of the particles, which are in the fracture at time zero, being transferred from the fracture to the matrix. C^{NE} decreases with time t monotonically.

In what follows, we will only discuss the particle-transfer probability corresponding to mass transfer from the fracture to the matrix and drop the subscript "*f*" and the superscript "*NE*" for simplicity. The derivation of the particle-transfer probability corresponding to mass transfer from the matrix to the fracture is similar and will not be repeated. For the particles that are in the fracture of a given grid cell at time = 0, we can write a mass conservation equation for those particular particles as follows:

$$C(0)(V_0 + K_d \rho_b V_T) = C(t)(V_0 + K_d \rho_b V_T) + F_{fm} \int_0 C(\tau)\, d\tau + F_{out} \int_0 C(\tau)\, d\tau \qquad \text{(Eq. 10)}$$

where V_0 and V_T are the volume of water in the fracture and the total volume of the fracture within the cell, respectively. ρ_b is the bulk density. F_{out} is defined as follows:

$$F_{out} = \sum_{i=1}^{M} \left[\max(Q_i, 0) + \frac{D_i A_i}{S_i} \right] \qquad \text{(Eq. 11)}$$

where M is the number of interfaces between the grid cell and other neighboring grid cells. Q_i (outward positive), D_i , A_i , and S_i are water flow rate, dispersion coefficient, area, and distance between the neighboring nodes of the i-th interface, respectively. The left-hand side of Equation 10 is the initial mass of the particles in the fracture of the given grid cell, while the first term of the right-hand side is the mass of the particles that still stay there at time t. The second term and the third term on the right-hand side of Equation 10 are the mass of particles flowing into the matrix of this cell and into the fracture continuum of other neighboring cells, respectively.

Taking derivatives on both sides of Equation 10 with respect to t, we have a first-order ordinary differential equation:

$$\frac{dC(t)}{dt} + \frac{1}{t_c} C(t) = 0 \qquad \text{(Eq. 12)}$$

with initial condition $C(0) = C_0$, where

$$t_c = \frac{V_0 + (1-\phi) K_d \rho_R V_T}{F_{fm} + F_{out}} \qquad \text{(Eq. 13)}$$

is the characteristic time in the system, which indicates how slowly the mass will be replaced for a given cell.

The solution of Equation 12 is readily obtained as:

$$C(t) = C_0 \exp(-t / t_c) \quad \text{(Eq. 14)}$$

Therefore, the probability of a particle being transferred from fracture to matrix during (0, t) can be calculated as:

$$P_{fm} = \frac{M_{fm}}{M_0} = \frac{F_{fm}\int_0^t C(\tau)\,d\tau}{C_0(V_0 + K_d \rho_b V_T)} = \frac{F_{fm}}{F_{out} + F_{fm}}\left[1 - \exp(-t / t_c)\right] \quad \text{(Eq. 15)}$$

Similarly, the particle-transfer probability corresponding to the mass flow from matrix to fracture, P_{mf}, can be calculated based on Equation 15 by replacing F_{fm} with F_{mf} and using F_{out} and t_c of the matrix continuum.

6.2.5 Adaptive Time Steps

Particle-tracking time steps used in DCPT are adaptive to the local flow field, cell size, and other transport parameters. For a given type of solute, each cell has two time steps corresponding to the fracture continuum and the matrix continuum, respectively. For either continuum of a cell, the time step is calculated as follows:

$$\Delta t = \min\left(0.05\frac{\Delta xy}{|Vxy|}, 0.05\frac{\Delta z}{|Vz|}, 0.25 t_c\right) \quad \text{(Eq. 16)}$$

where Δxy and $|Vxy|$ are the lateral size of the cell and the magnitude of the lateral component of the pore velocity within the cell, respectively, and Δz and $|Vz|$ are the height of the cell and the magnitude of the vertical component of the pore velocity, respectively. Equation 16 limits the time step so that the Courant Number (defined by $Co = \frac{\Delta t|Vxy|}{\Delta xy} or \frac{\Delta t|Vz|}{\Delta z}$ for the horizontal or vertical direction) is equal to or less than 0.05. This limit is sufficient for the proper accuracy of the explicit approaches used in DCPT by establishing an adequate temporal resolution regarding particle transfer between fracture and matrix. If sorption exists, effective velocities are used in Equation 16 by multiplying the original pore velocities by the factor P_r (see Section 6.2.2).

6.3 DESCRIPTION OF ADVECTIVE-DISPERSIVE SOLUTIONS WITH T2R3D

As a member of the TOUGH2 family of codes, T2R3D (Wu et al. 1996, pp. 8-32) provides a capability for modeling liquid or gas tracer or radionuclide transport in multiphase and nonisothermal flow systems. In particular, T2R3D can be used to simulate tracer transport in a complex, heterogeneous fractured rock using a general, irregular 3-D grid. In addition to incorporation of a full dispersion tensor in evaluating dispersive tracer transport, the code takes into account linear adsorption and first-order decay effects. The model formulation and numerical

scheme make it easy to include many transport mechanisms, such as nonadsorption, multidecay chains, or thermal/mechanical effects. T2R3D is built on the framework of the TOUGH2 code (Pruess 1991, pp. 5-9). The basic mass and thermal-energy balance equations for three-component fluid and heat solved by T2R3D are similar in form to those for the standard TOUGH2 EOS3 module (Pruess 1991, pp. 21-23). The integral finite-difference method and a first-order, backward finite-difference scheme are used for spatial and temporal discretization, respectively. The tracer transport mechanisms include molecular diffusion and hydrodynamic dispersion in the liquid or gaseous phase, in addition to advection terms. First-order decay is taken into account, and adsorption of a tracer on rock matrix and/or fractures is described by an equilibrium isotherm with a constant sorption distribution coefficient.

The model formulation considers advection/dispersion transport processes of a liquid or gas tracer with a full-dispersion tensor, in a heterogeneous geological system. The grid can be either regular or irregular. In addition to advection terms for the tracer transport, the dispersive and diffusive mass flux, $\mathbf{F}_D^{(\kappa)}$, is described by:

$$\mathbf{F}_D^{(\kappa)} = -\rho_\beta \bar{D} \bullet \nabla X_\beta^{(\kappa)} \qquad \text{(Eq. 17)}$$

where $\bar{D}$ is the combined diffusion-dispersion tensor accounting for both molecular diffusion and hydrodynamic dispersion; ρ_β is fluid density; and X_β is mass fraction of the tracer in phase β (β = liquid or gas) and superscript κ represents the solute component. A general dispersion model

for 3-D tracer transport in T2R3D is:

$$\bar{D} = \alpha_T |\mathbf{v}_\beta| \delta_{ij} + (\alpha_L - \alpha_T) \frac{\mathbf{v}_\beta \mathbf{v}_\beta}{|\mathbf{v}_\beta|} + \phi S_\beta \tau d_m \delta_{ij} \quad \text{(for } \beta\text{=liquid or gas)} \qquad \text{(Eq. 18)}$$

where α_T and α_L are the transverse and longitudinal dispersivities, respectively; $\mathbf{v}_\beta$ is the Darcy velocity vector of phase β through fractures or matrix; τ is the tortuosity of the medium; d_m is the molecular diffusion coefficient in phase β; and δ_{ij} is the Kronecker delta function ($\delta_{ij} = 1$ for $i = j$, and $\delta_{ij} = 0$ for $i \neq j$).

One of the key issues in implementing the general 3-D dispersion tensor of Equation 18 is how to interpolate velocity fields for determining the dispersion tensor. The averaging or weighting scheme used to evaluate a velocity vector at the interfaces between cells is called "*projected area weighting method*" (Wu and Pruess 1998, pp. 139-146). In this method, we calculate a velocity component, $v_{n,i}$, of the velocity vector of cell n by the summation of the flow components of all local connection vectors in the same direction, weighted by the projected area in that direction:

$$v_{n,i} = \frac{\sum_m (A_{nm} |\mathbf{n}_i|)(v_{nm} \mathbf{n}_i)}{\sum_m (A_{nm} |\mathbf{n}_i|)} \quad \text{(for i = x,y,z)} \qquad \text{(Eq. 19)}$$

where M is the total number of connections between cell n and all its neighboring cells, v_{nm} is the flux along the connection to cell m in the local coordinate system, and $\mathbf{n}_i$ are the directional cosines of connections. The velocity vector v at the interface of cells n and m is then evaluated by harmonic weighting to preserve total transit time for solute transport traveling between the two cells.

The mass fraction gradient of the tracer/radionuclide is evaluated at the interface between cells n and m as:

$$\nabla \mathbf{X}_{nm}^{(\kappa)} = \left(n_x \Delta X_{nm}^{(\kappa)}, n_y \Delta X_{nm}^{(\kappa)}, n_z \Delta X_{nm}^{(\kappa)}\right) \quad \text{(Eq. 20)}$$

with

$$\Delta X_{nm}^{(\kappa)} = \frac{X_m^{(\kappa)} - X_n^{(\kappa)}}{D_m + D_n} \quad \text{(Eq. 21)}$$

The net mass flux of diffusion and dispersion of a tracer/radionuclide along the connection of cells n and m is determined by Equation 17.

In the above calculation, the connection to the overlapping cell in the other continuum is excluded because it involves the mass transfer between the fracture continuum and the matrix continuum. This mass transfer is treated as a 1-D advection-dispersion transport process and added to the mass conservation equation of each cell.

Though T2R3D has this capability, radioactive decay is not used in this comparison because the effect of radioactive decay would be essentially the same for all of the methods being compared here.

6.4 COMPARISONS OF FEHM AND DCPT WITH T2R3D

In this section, DCPT is first compared with analytical solutions for 1-D and 2-D cases in Sections 6.4.1 and 6.4.2. The FEHM particle-tracking code has been previously compared to analytical solutions as part of its qualification (FEHM, STN:10031-2.00-00, Version 2.0). Both the DCPT and FEHM particle tracker are compared with T2R3D for a 1-D case in Section 6.4.3, and then the DCPT is compared with T2R3D for the full 3-D case of the Yucca Mountain UZ Model.

Again, the FEHM particle-tracking code was used in TSPA-VA. Radioactive decay is not included in all of the comparisons discussed below because the effect of radioactive decay would be essentially the same for all of the methods being compared here.

The numerical values of physical and geometric parameters for the selected test cases were chosen to provide reasonable representations of the real-world scales and properties appropriate to the flow and transport process under consideration.

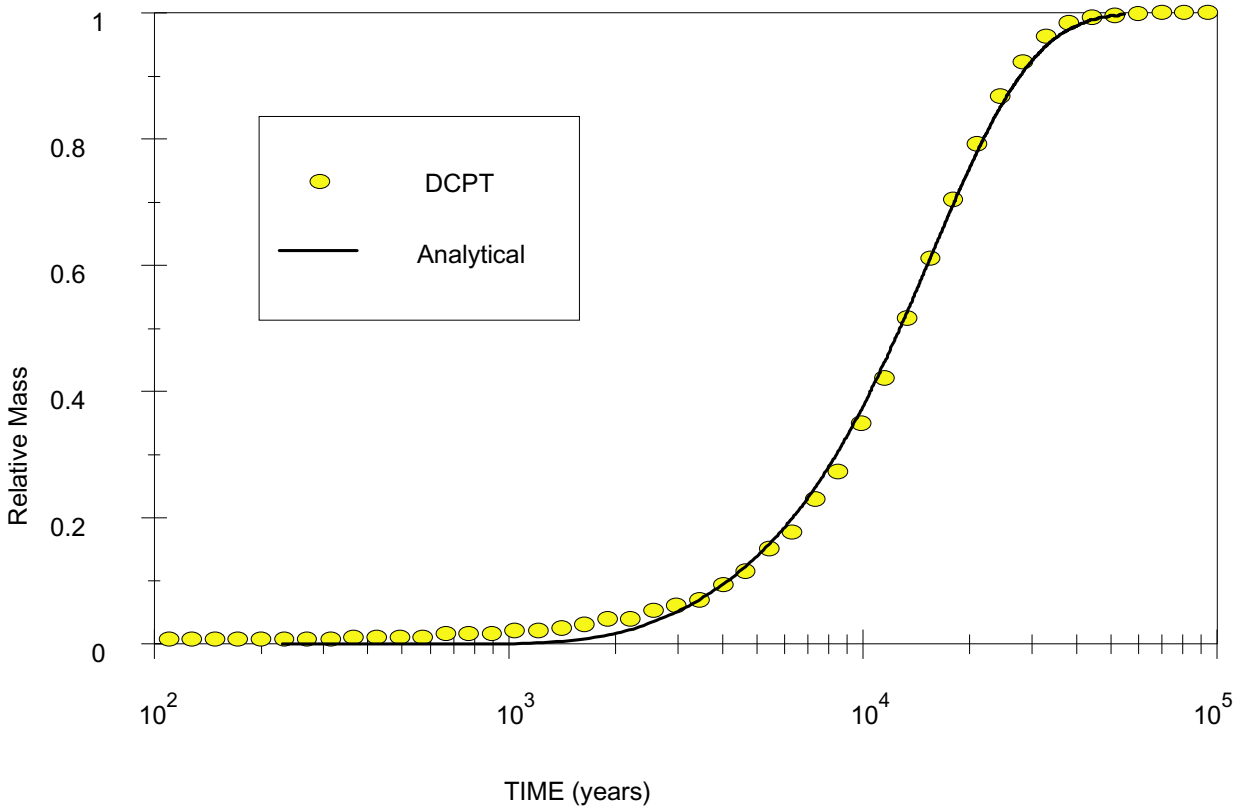
6.4.1 1-D Cases Comparing Analytical Solutions with DCPT

The first test case is 1-D solute transport in a fractured-porous medium with parallel fractures, for which the particle-transfer-probability approach and the sorption model of DCPT can be tested against an analytical solution (Sudicky and Frind 1982, pp. 1634-1642). The analytical solution is based on the approximation that solute transport between fractures and matrix occurs through matrix diffusion in the direction perpendicular to the fracture only. Matrix advection and diffusion in the direction along the fracture is ignored. Furthermore, the initial solute concentration is zero in the system, and the concentration at inlets of fractures ($z = 0$) is constant for time $t > 0$. The diffusion/dispersion in the fractures is also ignored. The rationale for the parameters shown in Table 4 is documented in Scientific Notebook YMP-LBNL-GSB-LHH-1, pp 83-89. For this case, the integral of the breakthrough curve corresponding to a pulse input is equivalent to the breakthrough curve corresponding to a constant concentration input, which is the solution in Sudicky and Frind (1982, pp. 1634-1642).

Table 4. Parameters Used in Transport Problem in a Parallel Fracture System

Parameter	Value
Molecular diffusion coefficient	2.5×10^{-11} m^2/s
Fracture spacing	1.0 m
Retardation factor	30
Velocity in fracture	1.1574×10^{-5} m/s
Grid spacing	0.5 m
Matrix volume per cell	0.25 m^3
Fracture volume per cell	0.5×10^{-4}
Fracture/matrix interface area	0.5 m^2
Domain length	36.75 m

Figure 1 shows the cumulative mass fraction (the integral of the DCPT breakthrough curve divided by the initial mass released) flowing out from the fracture at the outlet as a function of time. The results from DCPT are similar to those for the analytical solution. This implies that the particle-transfer-probability approach (used in DCPT) of diffusive mass exchange between fracture and matrix is representative for this transient case. Note that the fracture spacing is 1.0 meter, which is within the range of the fracture spacing in the unsaturated zone of the Yucca Mountain site. The CPU time used in simulation by DCPT on a PC (Pentium II 300) is about 10 seconds, excluding the time used for reading/writing files. Filenames are given in Attachment II.



Based on data submitted with this AMR under DTN: LB990901233129.001

Figure 1. Comparison between DCPT and the Analytical Solution for a Parallel Fracture System

6.4.2 2-D Cases Comparing Analytical Solutions with DCPT

The second test case is 2-D solute transport in a porous medium (no fractures) with a dispersion tensor, for which the advection and dispersion model of DCPT can be tested against an analytical solution. Table 5 shows the case specifications with all parameters dimensionless; the rationale for these parameters is documented in Scientific Notebook YMP-LBNL-GSB-LP-3, pp. 1-105.

Table 5. Parameters of the 2-D Case

Parameters	Value
Domain dimension (x, y, z)	20.5×20.5×30.5
Pore velocity	Vx = Vy = 0, Vz = 1
Dispersivity	α_L = 0.05, α_T = 0.01
Diffusion coefficient	0.0
Grid spacing	Δx = Δz = 0.5; Δy = 20.5
Plume location at t = 0	x = 10.25, y = 10.25, and z = 0.0
Monitoring location at t = 10	x = 10.25
Monitoring resolutions	δx = 0.02 and δz = 0.01

As defined in Table 5, the scenario is a simultaneous injection of mass at time zero in a 2-D uniform flow field (flow in z-direction only). If M is the mass of a point source injected at (x_0, z_0) at $t = 0$, the concentration distribution in the field at any later time is given by (Bear 1972, p. 633, Equation 10.6.34, symbolically replacing n, ξ, η, D', D'', and q/n with ϕ, z_0, x_0, D_z, D_x, and V_z):

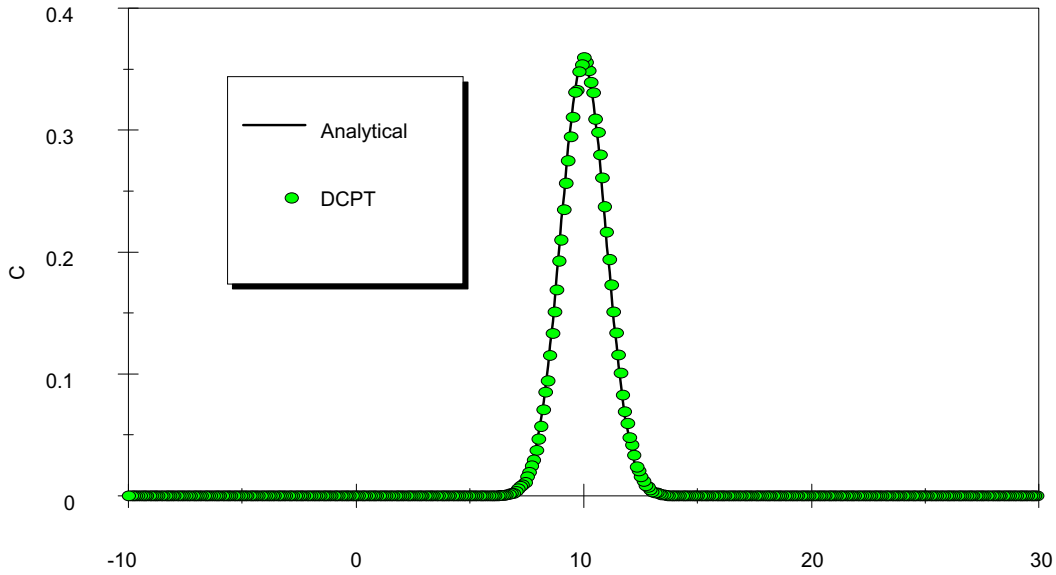
$$C(x, z, t) = \frac{\mathrm{M}/\phi}{4\pi t\sqrt{D_x D_z}} \exp\left[-\frac{(z - z_0 - V_z t)^2}{4D_z t} - \frac{(x - x_0)^2}{4D_x t}\right] \qquad \text{(Eq. 22)}$$

where D_x (= $\alpha_T V_x$) and D_z (= $\alpha_L V_z$) are dispersion coefficients corresponding to x-direction and z-direction, respectively, and ϕ is porosity. The problem is actually simulated with DCPT as a 3-D transport problem with no discretization in the y-direction. Solutes are released at time zero in the form of a point pulse source (M/ϕ = 1). At t = 10, the relative concentration along $x = x_0$ (= 10.25) is calculated within the specific slice. Figure 2 compares these results with the analytical solution. The concentration distribution simulated by DCPT is consistent with the analytical solution. This consistency indicates that DCPT properly incorporated the dispersion tensor.

All values are dimensionless.

Two million particles were used in the simulation (Figure 2), and the CPU time used is about 10 minutes on a PC with a Pentium II 300 processor.

All input and output filenames are given in Attachment II, Section 1.



Based on data submitted with this AMR under DTN: LB990901233129.001

Figure 2. Comparison between DCPT and the Analytical Solution for a 2-D Transport Problem with Dispersion Tensors.

6.4.3 Comparison of Numerical Solutions (T2R3D) with FEHM and DCPT for 1-D Cases

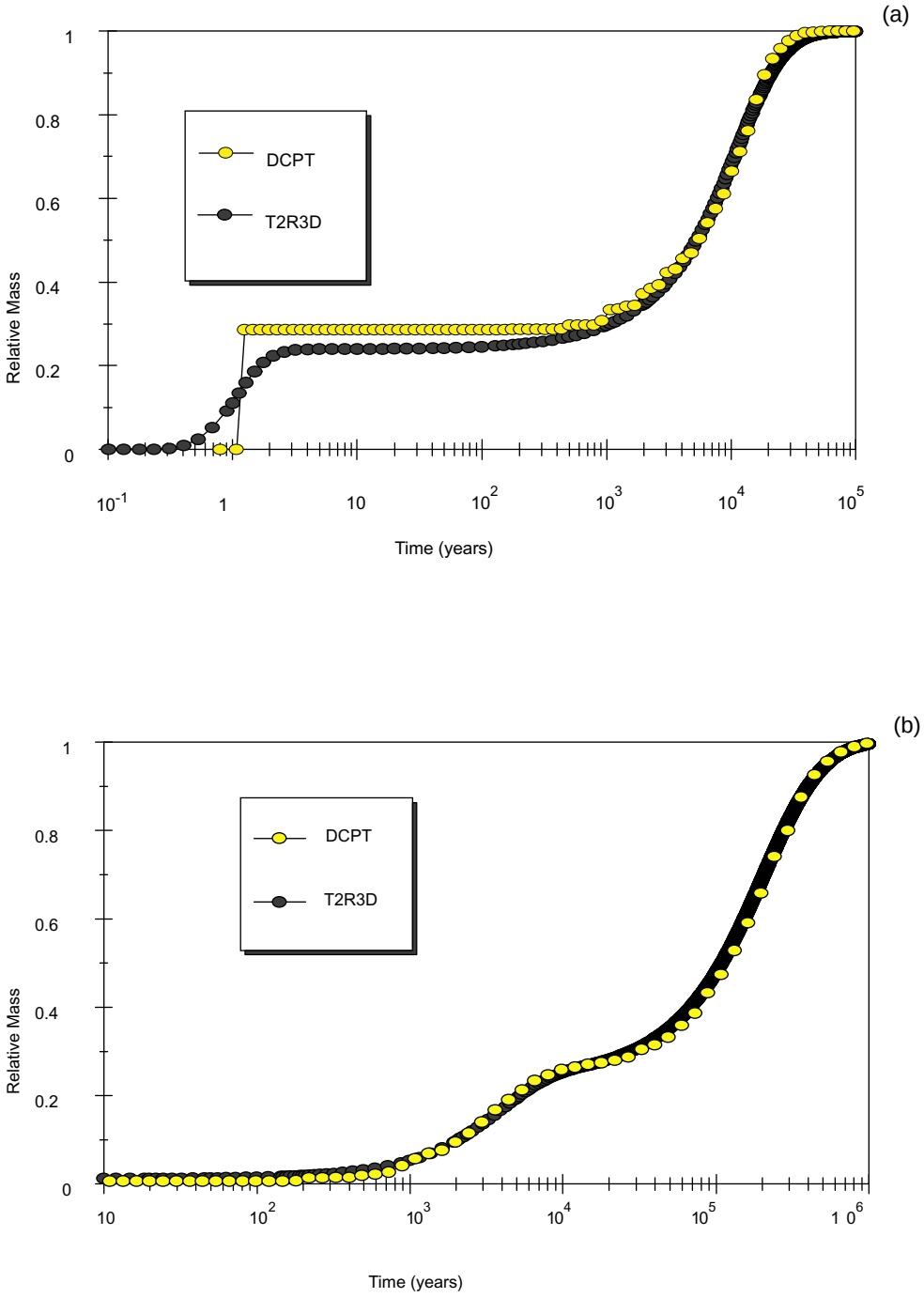
Analytical solutions are only available for the simplified cases (e.g., no advective flow between fracture and matrix). In those cases, the critical features of the particle-tracking models cannot be fully tested. A more realistic one-dimensional transport problem is thus designed to further test the capabilities of the particle-tracking models against the numerical solutions provided by T2R3D, mainly focusing on simulations of the fracture-matrix mass exchange and sorption processes. The case involves a column near borehole USW SD-9 extracted from the 1997 3-D model of the Yucca Mountain site (DTN: LB971212001254.001). The radionuclides are released at the simulated repository horizon at time zero as a pulse. A steady-state flow field is assumed and determined using TOUGH2 version 1.4. The transport parameters used in simulations are shown in Table 6. A total of 2,000 particles are used in DCPT simulation. The CPU time used is about 10 seconds for both DCPT and T2R3D, with DCPT executed on a Pentium II PC and T2R3D on a DEC ALPHA. A total of 27 cells are used.

Table 6. Parameters Used for 1-D Radionuclide Transport

Parameter	Value
Molecular diffusion coefficient of technetium	3.2×10^{-11} m^2/s
Molecular diffusion coefficient of neptunium	1.6×10^{-10} m^2/s
Fracture longitudinal and transverse dispersivity	20 m and 0
Matrix longitudinal and transverse dispersivity	0
Fracture-matrix dispersivity	0
Fracture and matrix tortuosities	0.7 and 0.7
Temperature	25 °C
Sorption distribution coefficients of technetium	Zero in both fracture and matrix
Sorption distribution coefficients of neptunium	Zero in fracture and matrix of TCw, PTn, TSw units; 4.0×10^{-3} m^3/kg and 1.0×10^{-3} m^3/kg in matrix of zeolitic rock and vitric rock in CHn unit, respectively.

In this case, significant mass flow occurs as a result of advection and dispersion between fracture and matrix. Figure 3 shows the cumulative mass fraction at the water table versus time. The cumulative mass fraction is defined as the cumulative mass flowing out to groundwater divided by the total mass released at the repository horizon. In both cases, the results are very similar except that the DCPT shows fewer numerical mixing effects than the T2R3D. The good agreements between the DCPT and the T2R3D show that the approximation (A = V) in Equation 2 is acceptable for the UZ transport of radionuclides in the Yucca Mountain site. The input and output files and the process for performing DCPT simulations are provided in Attachment II, Section 1.

The comparison of the FEHM particle-tracking simulations with the advective-dispersive (A-D) transport simulations of T2R3D consisted of a single 1-D flow simulation along borehole USW SD-9, with four subsequent transport simulations. The details regarding input and ouput files and use of software macros for this part of the analysis are provided in Attachment II, Section 2 (DTN: SN9908T0581699.001). The four transport simulations are detailed in Table 7.



Based on data submitted with this AMR under DTN: LB990901233129.001

Figure 3. Comparison between DCPT and T2R3D for 1-D Radionuclide Transport. (a) Technetium, (b) Neptunium

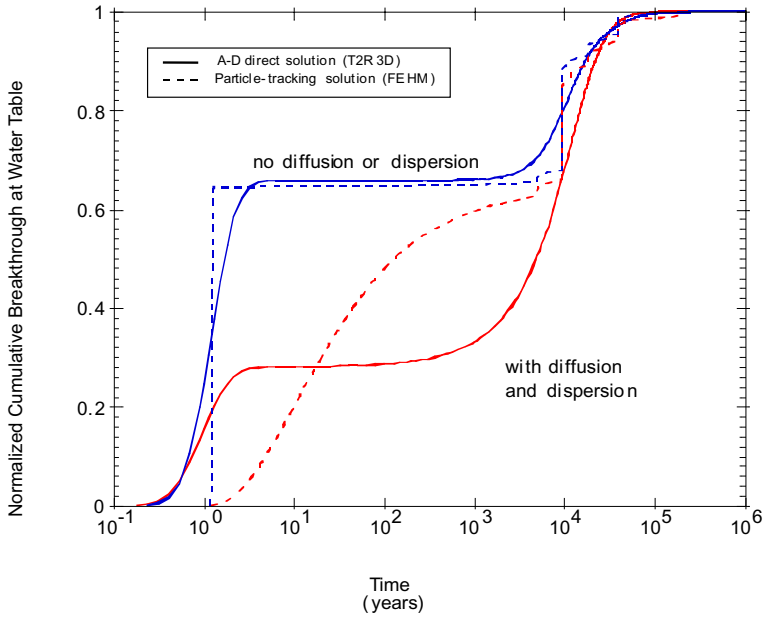
Table 7. Four Transport Simulations Used in FEHM vs. T2R3D Comparison

Radionuclide Simulated	Molecular Diffusion (m^2/s)	Distribution Coefficient, K_d (m^3/kg) in Vitric, Zeolitic	Fracture Dispersivity (m)
Technetium (Tc)	3.2×10^{-11}	0, 0	20
Technetium (Tc)	0	0, 0	0
Neptunium (Np)	1.6×10^{-10}	1.0×10^{-3}, 4.0×10^{-3}	20
Neptunium (Np)	0	1.0×10^{-3}, 4.0×10^{-3}	0

These four simulations consider the transport of two radionuclides, Tc and Np, under conditions with and without matrix diffusion and fracture dispersivity. The Np is assumed to sorb within the matrix, but the Tc does not, and in no case does sorption occur along the fracture. The sorption distribution coefficients for the matrix of different geological units are given in Table 6. Particularly, the thickness of the vitric rock ($K_d=1\times10^{-3}$ m^3/kg) and the zeolithic rock ($K_d=4\times10^{-3}$ m^3/kg) in CHn unit is 46.63m and 103.16 m, respectively. A finite amount of radionuclides was released at a cell near the potential repository elevation at 1063 m, and the transport simulation was run for one million years, with the cumulative breakthrough (normalized) of the radionuclide plotted as a function of time for each of the four cases.

CPU time used for each simulation using FEHM particle tracker is less than 1 minute on a SUN ULTRA SPARC machine.

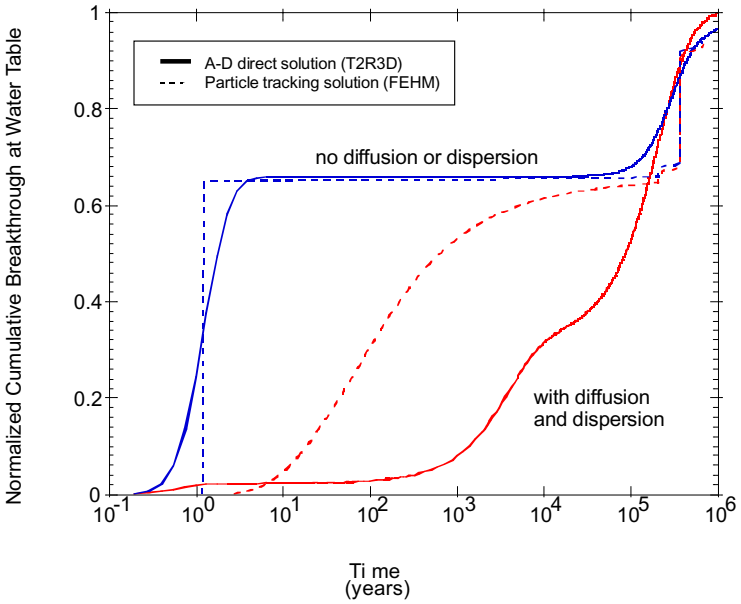
The results of the simulations are shown in Figures 4 and 5. Figure 4 shows the cumulative normalized breakthrough for technetium. The solid lines are the results of T2R3D; the dashed lines are the results of FEHM. Both cases, with and without matrix diffusion and sorption, are shown. Results for T2R3D and FEHM are very similar for the advection-only case. The FEHM particle-tracking results show sharper breakthrough fronts at the water table. This is reasonable because the particle-tracking method reduces the numerical dispersion associated with finite-difference and finite-element methods as used in T2R3D. The initial breakthrough at around one year is a result of advective transport of technetium through the fractures. Both methods show that over 60% of technetium reaches the water table in the case without matrix diffusion and dispersion. The second major breakthrough in the case without dispersion or matrix diffusion occurs around 10,000 years. This breakthrough represents the transport through the matrix continuum between the repository and water table.



Based on data submitted with this AMR under DTN: SN9908T0581699.001

NOTE: Diffusion Coefficient = 3.2 x 10^{-11} m^2/s, dispersivity = 20 m (fractures only), K_d = 0 m^3/kg

Figure 4. Normalized Cumulative Breakthrough of Technetium at the Water Table for FEHM and T2R3D



Based on data submitted with this AMR under DTN: SN9908T0581699.001

NOTE: Diffusion Coefficient = 1.6 x10^{-10} m^2/s, dispersivity = 20 m (fractures only), K_d = 4.0 x 10^{-3} m^3/kg (zeolitic), K_d= 1.0 x 10^{-3} m^3/kg (vitric)

Figure 5. Normalized Cumulative Breakthrough of Neptunium at the Water Table for FEHM and T2R3D

In Figure 4, the technetium transport with matrix diffusion is significantly different in the T2R3D and FEHM simulations. The FEHM results indicate that the initial breakthrough in the fractures is "smeared," but the asymptotic plateau is the same as the plateau for the case with no matrix diffusion (~65%). The reason is that the implementation of the diffusive mass flow from the fracture to the matrix in the FEHM particle-tracking algorithm yields additional residence time (a retardation) for the particles in the fractures that experience diffusive mass flow from fracture into matrix, but the particles do not actually get transported into the matrix. As a consequence, the shape of the initial breakthrough for the FEHM simulation yields the same plateau as the case with no matrix diffusion. This approach is based on an analytical solution by Tang et al. (1981, pp. 555-564), which assumes that diffusive mass flow within the matrix only occurs in the direction perpendicular to the fracture. Hence, the particles can leave the flow system via the fracture only. However, like the DCPT and the T2R3D, the FEHM particle tracker implements the advective mass flow in both fracture and matrix continua, which allows the particles to transport to the water table through either fracture or matrix. As a result, the FEHM particle tracker gives very similar results to those of T2R3D for the advection-only cases.

The T2R3D results, in contrast, show an initial breakthrough that has less than 30% of technetium arriving at the water table through the fractures. Recall that without the diffusion between fracture and matrix (which is controlled by the matrix diffusion), over 60% of the technetium arrived at the water table through the fractures. The balance is transported into the matrix via the diffusive mass flow according to the method used in T2R3D. Once inside the matrix, the radionuclide can be advected through the matrix only at a much slower rate than through the fractures unless it transports into the fracture again at some later time either by advection or dispersion. This approach apparently yields a slower overall transport to the water table than the FEHM simulation, which adds additional residence time to particles in fracture elements rather than allowing them to actually transport into the matrix via the diffusive mass flow. The median breakthrough time of radionuclides with matrix diffusion is about 100 years for the FEHM simulation and several thousand years for the T2R3D simulation.

Similar results are obtained in Figure 5, which shows the normalized breakthrough of neptunium at the water table for the FEHM and T2R3D simulations. These simulations include sorption in the vitric and zeolitic matrix elements. The runs with no matrix diffusion or dispersion show little difference in the initial breakthrough of neptunium except for the sharpness of the front, similar to the technetium simulations. For neptunium, however, the secondary breakthrough is delayed past 100,000 years because of sorption in the matrix. The runs with matrix diffusion show a disparity between the results of T2R3D and FEHM that are similar to the technetium runs. The reasons are the same for both radionuclides, but the difference is even more pronounced when sorption occurs in the matrix. The median breakthrough time for neptunium is nearly a thousand years for the FEHM simulation, but it is nearly 100,000 years for the T2R3D simulation.

These results indicate that a significant difference exists in representations of the diffusive mass flow between fracture and matrix in FEHM and T2R3D. The diffusive mass flow between the fracture and matrix model in T2R3D allows the radionuclides to diffuse into the matrix, yielding much lower initial breakthrough via the fractures. FEHM results are based on an analytical solution that accounts for transient gradients in the matrix (though not valid for the finite matrix and the flow field here), but the absence of radionuclide transport into the matrix via diffusion is

less consistent with the dual-permeability formulation used in the flow simulations. The input and output filenames associated with these runs are described in Attachment II.

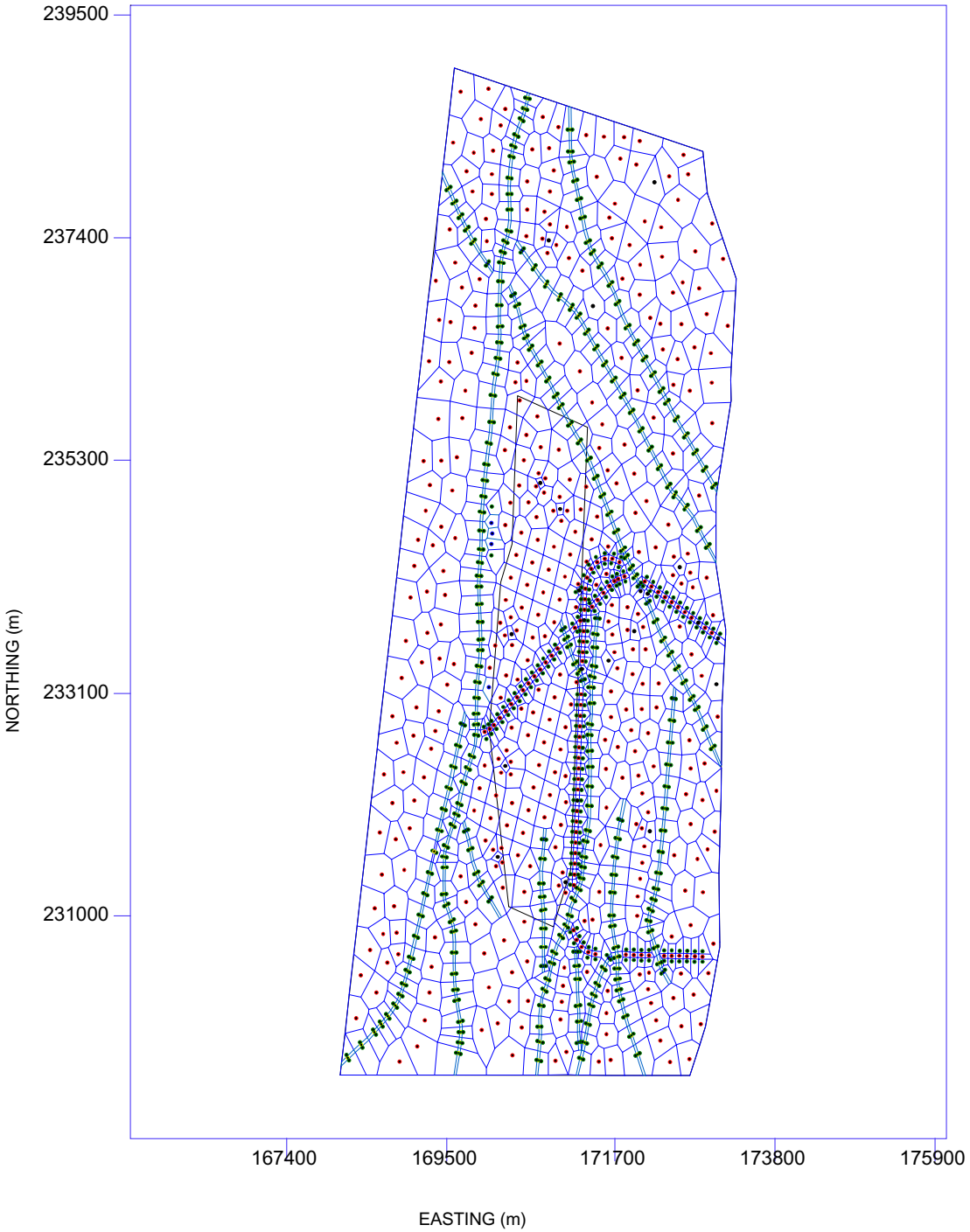
6.4.4 Comparison of Numerical Solutions (T2R3D) with DCPT for Full 3-D Model of Yucca Mountain Site

The full 3-D model of the Yucca Mountain unsaturated zone is a comprehensive, mountain-scale model. It includes all known aspects of flow and transport processes in the fractured-porous media, and provides a comprehensive test case for the particle-tracking simulator and other numerical simulators. Comparison of the particle tracker (DCPT) with the numerical solutions (T2R3D) provides insights into these methods for a complex system.

A comparison of FEHM particle-tracker with DCPT for full 3-D model of Yucca Mountain Site can be found in AMR U0160 (CRWMS M&O 2000a, Section 6.2.4, pp. 21–22). That comparison shows discrepancies similar to those found in Section 6.4.3 of this report.

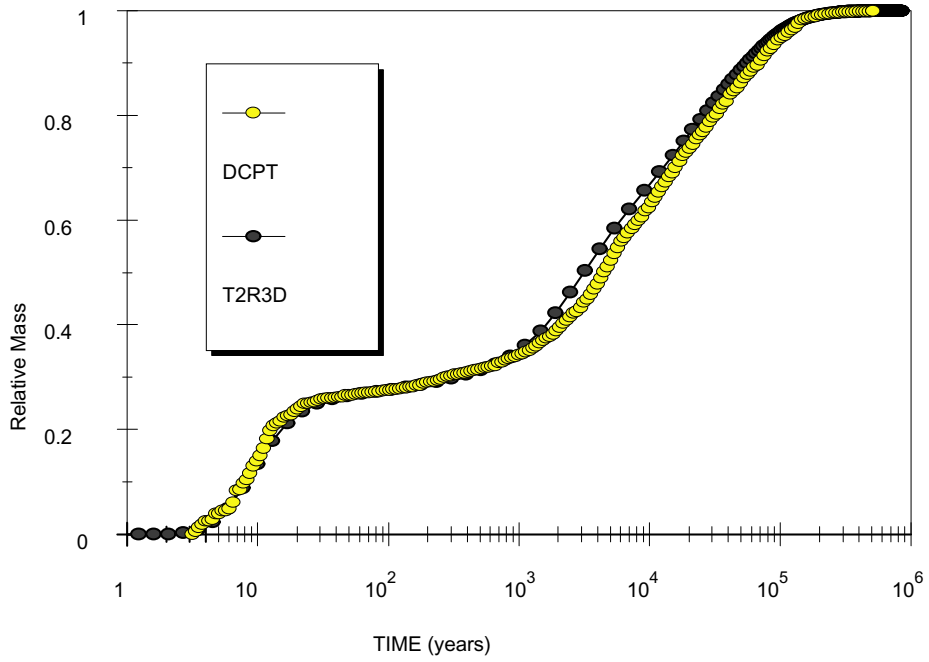
Figure 6 shows a plan view of the 3-D grid. For these simulations, the radionuclides are released at the simulated repository horizon with time zero as a pulse. Steady-state flow is calculated using TOUGH2 V1.4 (TOUGH2 V1.4, STN: 10007-1.4-00, Version 1.4) with hydraulic properties in DTN: LB997141233129.001. The transport parameters are the same as those in Table 5. A total of 1,680 particles are used in the simulations using DCPT. The corresponding CPU time used for each run is about 20 seconds using DCPT on a Pentium II PC and about 1 hour using T2R3D on a DEC ALPHA.

The cumulative mass fractions entering groundwater versus time are depicted on Figures 7 (technetium) and 8 (neptunium). The results of DCPT agree very well with the results of T2R3D. This argument implies that DCPT can provide results nearly identical to those of T2R3D, which rigorously solves the advection-dispersion equation of radionuclide transport in the Yucca Mountain site. Its performance will not diminish as the size of the grid (number of cells) increases, a feature that is particularly important in large-scale models such as the UZ Model of for Yucca Mountain.



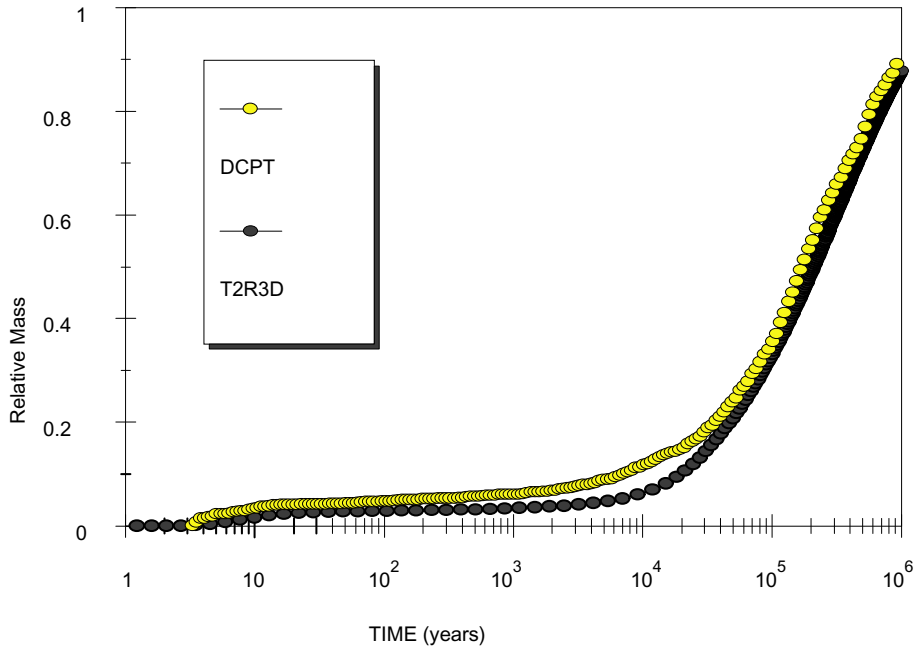
Based on data from DTN: LB990501233129.004

Figure 6. Map View of the 3-D Grid



Based on data submitted with this AMR under DTN: LB990901233129.001

Figure 7. Comparison between DCPT and T2R3D for 3-D Radionuclide Transport of Technetium



Based on data submitted with this AMR under DTN: LB990901233129.001

Figure 8. Comparison between DCPT and T2R3D for 3-D Radionuclide Transport of Neptunium

INTENTIONALLY LEFT BLANK

7. CONCLUSIONS

Different methods for simulating radionuclide transport in unsaturated, fractured media were compared under conditions consistent with those expected at Yucca Mountain. These comparisons utilized 1-D and 3-D flow fields developed using the UZ Model, a dual-continua model calibrated to hydrologic conditions at Yucca Mountain. The methods compared included two particle-tracking methodologies, FEHM and DCPT, and one integral finite-difference method, T2R3D, which utilizes a fully coupled advective-dispersive solution. The latter method is considered to be a more rigorous approach, but is not always appropriate for large-scale problems because of its computational requirements. The modeling results reported in this AMR have been submitted to the TDMS under DTN: LB990901233129.001 and DTN: SN9908T0581699.001.

The advantage of using a particle-tracking model (DCPT or FEHM) over a fully coupled advective-dispersive simulator (T2R3D) would be in its computational efficiency and lower CPU requirement, with less numerical diffusion in the case of small physical diffusion coefficients. The comparisons of T2R3D and DCPT revealed that DCPT provides results nearly identical to those of T2R3D for the time frames and scenarios considered. It can effectively simulate complex transport processes of radionuclides in dual-continua media. It is an efficient simulator, in terms of computational requirements, especially when only a cumulative breakthrough curve is required. Its performance will not diminish as the number of the grid cells increases, a feature that is of particular importance in large-scale models. Additionally, the DCPT provides higher spatial resolution since it allows particles to move through a continuous space.

One-dimensional comparisons performed using the FEHM particle-tracking method and T2R3D indicated that the two methods agree only if diffusion and dispersion are neglected. For the cases that include diffusion and dispersion, the median breakthrough for FEHM occurred at times more than one to two orders of magnitude earlier than the simulations for T2R3D for the scenarios considered. This difference resulted from the use of a residence-time-transfer function to account for the effects of the diffusive mass flow between the fracture and the matrix in FEHM. Particles advected and dispersed in the fracture continuum are modeled as if they remain along these fast flow paths, and the residence-time-transfer-function algorithm is utilized to adjust particle residence times to reflect the time lag attributed to diffusion into and out of the matrix. This difference between T2R3D and FEHM is more pronounced for radionuclides undergoing sorption in the matrix. Numerical experiments reveal that the diffusive mass flow between fractures and the matrix is one of the key processes that control the travel time of radionuclides to water table in the Yucca Mountain, even though the dispersion processes in either fractures or the matrix have little effect.

This notable difference in the results for the particle-tracking methods stems from different implementations of the diffusive mass flow between fractures and the matrix in the two codes. Essentially, as noted above, FEHM utilizes a residence-time-transfer function in accounting for diffusion into matrix, resulting in a formulation less consistent with the dual-permeability approach. As a result, the total mass flow from the fracture into the matrix is underestimated relative to a fully coupled advective-dispersive solution. With DCPT, both advection and dispersion/diffusion are incorporated simultaneously into the particle-transfer probability, providing an approach more consistent with the dual-permeability approach. As such, the DCPT

is a better suited particle-tracking methodology than FEHM for a dual-continua model with a structure similar to that of the UZ Model.

For a 10,000-year period, particle tracking using FEHM produces more conservative results by overpredicting the mass of radionuclides that will reach the water table. FEHM has already been used for transport simulations in the TSPA-VA, and past results should be considered conservative given the analysis presented here. Continued use of this code would not underestimate risk and, therefore, would not be invalid from a federal or state regulatory viewpoint. Its use, though, will underestimate the performance of the unsaturated zone as a barrier to radionuclide transport. Utilizing DCPT or T2R3D or similar approaches possibly implemented in FEHM for TSPA calculations would result in better calculated performance of the unsaturated zone, potentially by orders of magnitude compared with FEHM.

8. INPUTS AND REFERENCES

8.1 DOCUMENTS CITED

Bear, J. 1972. *Dynamics of Fluid in Porous Media*. New York, New York: Dover Publications. TIC: 217568.

CRWMS M&O (Civilian Radioactive Waste Management System, Management & Operations Contractor) 1999a. *Analysis and Modeling Report Development Plan (DP) for U0155 Analysis Comparing Advective-Dispersive Transport Solution to Particle-tracking, Rev. 00.* TDP-NBS-HS-000002. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990826.0106.

CRWMS M&O 1999b. *M&O Site Investigations.* Activity Evaluation. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990317.0330.

CRWMS M&O 1999c. *M&O Site Investigations.* Activity Evaluation. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990817.0257.

CRWMS M&O 1999d. *Performance Assessment Operations.* Activity Evaluation. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990716.0106.

CRWMS M&O 2000a. *Analysis of Base-Case Particle Tracking Results of the Base-Case Flow Fields.* ANL-NBS-HS-000024. Las Vegas, Nevada: CRWMS M&O.

CRWMS M&O 2000b. *Particle Tracking Model and Abstraction of Transport Processes.* ANL-NBS-HS-000026. Las Vegas, Nevada: CRWMS M&O. URN-0037

Doughty, C. 1999. "Investigation of Conceptual and Numerical Approaches for Evaluating Moisture, Gas, Chemical, and Heat Transport in Fractured Unsaturated Rock." *Journal of Contaminant Hydrology, 38* (1-3), 69-106. Amsterdam, The Netherlands: Elsevier Science Publishers. TIC: 244160.

Dyer, J.R. 1999. "Revised Interim Guidance Pending Issuance of New U.S. Nuclear Regulatory Commission (NRC) Regulations (Revision 01, July 22, 1999), for Yucca Mountain, Nevada." Letter from J.R. Dyer (DOE) to D.R. Wilkins (CRWMS M&O), September 9, 1999, OL&RC:SB-1714, with enclosure, "Interim Guidance Pending Issuance of New U.S. Nuclear Regulatory Commission (NRC) Regulations (Revision 01)." ACC: MOL.19990910.0079.

LaBolle, E.M.; Fogg; G.E.; and Tompson, A.F.B. 1996. "Random-Walk Simulation of Transport in Heterogeneous Porous Media: Local Mass-Conservation Problem and Implementation Methods." *Water Resources Research, 32* (4), 583-593. Washington, D.C.: American Geophysical Union. TIC: 245563.

Pruess, K. 1991. TOUGH2 – *A General Purpose Numerical Simulator for Multiphase Fluid and Heat Flow.* Report LBL-29400. Berkeley, California: Lawrence Berkeley National Laboratory. ACC: NNA.19940202.0088.

Sudicky, E.A. and Frind, E.O. 1982. "Contaminant Transport in Fractured Porous Media: Analytical Solutions for a System of Parallel Fractures." *Water Resources Research, 18* (6), 1634-1642. Washington, D.C.: American Geophysical Union. TIC: 217475.

Tang, D.H.; Frind, E.O; and Sudicky, E.A 1981. "Contaminant Transport in Fractured Porous Media: Analytical Solution for a Single Fracture." *Water Resources Research, 17* (3), 555-564. Washington, D.C.: American Geophysical Union. TIC: 225358.

Wu, Y.S.; Ahlers, C.F.; Fraser, P.; Simmons, A.; and Pruess, K. 1996. *Software Qualification of Selected TOUGH2 Modules*. Report LBNL-39490. Berkeley, California: Lawrence Berkeley National Laboratory. ACC: MOL.19970219.0104.

Wu, Y. S. and K. Pruess. 1998. "A 3-D Hydrodynamic Dispersion Model for Modeling Tracer Transport in Geothermal Reservoirs." *Proceedings, Twenty-third Workshop, Geothermal Reservoir Engineering, Stanford, California, January 26-28, 1998,* 139-146. Stanford, California: Stanford University. TIC: 245292.

Wu, Y.S.; Haukwa, C. and Bodvarsson, G.S. 1999. "A Site-Scale Model for Fluid and Heat Flow in the Unsaturated Zone of Yucca Mountain, Nevada." *Journal of Contaminant Hydrology, 38* (1-3), 185-215. Amsterdam, The Netherlands: Elsevier Science Publishers. TIC: 244160.

SOFTWARE CITED

Software Code: TOUGH2 V.1.4, STN: 10007-1.4-01.

Software Code: T2R3D V.1.4, STN: 10006-1.4-00.

Software Code: DCPT V.1.0, STN: 10078-1.0-00.

Software Code: FEHM V.2.0, STN: 10031-2.00-00.

Macro/Routine: MAKEPTRK V.1.0. ACC: MOL.19990915.0361.

Macro/Routine: PROCESS1 V.1.0. ACC: MOL.19990915.0360.

Macro/Routine: T2FEHM2 V.2.0. ACC: MOL.19990915.0359

Macro/Routine: PrepareKDfile V1.0. ACC: MOL.20000127.0120.

Macro/Routine: ExtractFlow V1.0. ACC: MOL.20000127.0121.

Macro/Routine: ExBT V1.0. ACC: MOL.20000127.0122.

Macro/Routine: StatSpatial V1.0. ACC: MOL.20000202.0193

8.2 STANDARDS, CODES, REGULATIONS AND PROCEDURES CITED

64 FR (Federal Register) 8640. Disposal of High-Level Radioactive Waste in a Proposed Geologic Repository at Yucca Mountain. Proposed rule 10 CFR 63. Readily available.

DOE (U.S. Department of Energy) 1998. *Quality Assurance and Requirements Description.* DOE/RW-0333P, REV 8. Washington D.C.: DOE OCRWM. ACC: MOL.19980601.0022.

QAP-2-3 *Classification of Permanent Items*. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990316.0006.

8.3 SOURCE DATA, LISTED BY DATA TRACKING NUMBER

LB971212001254.001. DKM Basecase Parameter Set for UZ Model with Mean Fracture Alpha, Present Day Infiltration, and Estimated Welded, Non-Welded and Zeolitic FMX. Submittal date: 12/12/1997.

LB990501233129.004. Mesh Files for 3-D UZ Model Calibration. Submittal date: 09/24/1999.

LB997141233129.001. Calibrated Basecase Infiltration 1-D Parameter Set for The UZ Flow and Transport Model, FY99. Submittal date: 07/21/1999.

8.4 AMR OUTPUT DATA LISTED BY DATA TRACKING NUMBER

LB99091233123.001. Modeling Results for DCPT Comparisons. Submittal date: 09/27/1999.

SN9908T0581699.001. Files to Support 1-D Comparison Between FEHM Particle Tracking and T2R3D Advective-Dispersive Transport Simulations along SD-9. Submittal date: 08/16/1999.

INTENTIONALLY LEFT BLANK

9. ATTACHMENTS

ATTACHMENT I - DOCUMENT INPUT REFERENCE SHEET

ATTACHMENT II - INPUT AND OUTPUT FILES FOR DCPT AND FEHM

1. FILES FOR DCPT

2. FILES FOR FEHM

ATTACHMENT III - SOFTWARE ROUTINES

INTENTIONALLY LEFT BLANK

ATTACHMENT I—DOCUMENT INPUT REFERENCE SHEET

DIRS as of the issue date of this AMR. Refer to the DIRS database for the current status of these inputs.

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev: ANL-NBS-HS-000001/00
Change:
Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking

Input Document			4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
	2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
1.	DTN: LB971212001254.001. DKM Basecase Parameter Set for UZ Model With Mean Fracture Alpha, Present Day Infiltration, and Estimated Welded, Non-Welded and Zeolitic FMX.	Parameters & mesh file	N/A – Reference only	6.4	Flow parameters for DKM Base-case, mean permeability, present day infiltration MESH file for present day infiltration	N/A	N/A	N/A	N/A
2.	DTN: LB990501233129.004. Mesh Files for 3-D UZ Model Calibration.	Entire	N/A-Technical Product Output	4	3-D Grid Mesh-filename	N/A	N/A	N/A	N/A
3.	DTN: LB997141233129.001. Calibrated Basecase Infiltration 1-D Parameter Set for the UZ Flow and Transport Model, FY99.	1dbasecaseR1 wodis.xls	N/A-Technical Product Output	6.4	Flow parameters for base-case, present infiltration	N/A	N/A	N/A	N/A
4.	Bear, J. 1972. *Dynamics of Fluid in Porous Media.* New York, New York: Dover Publications. TIC: 217568.	p. 70 p. 633	N/A - Reference only	6.2.1 6.4.2	Equation 4.1.18 Equation 10.6.34	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
5. CRWMS M&O (Civilian Radioactive Waste Management System, Management & Operations Contractor) 1999a. *Analysis and Modeling Report Development Plan (DP) for U0155 Analysis Comparing Advective-Dispersive Transport Solution to Particle-tracking, Rev. 00.* TDP-NBS-HS-000002. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990826.0106.	Entire	N/A - Reference only	2	Planning Document	N/A	N/A	N/A	N/A
6. CRWMS M&O 1999b. *M&O Site Investigations.* Activity Evaluation. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990317.0330.	Entire	N/A - Reference only	2	Activity Evaluation	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00			Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document			4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
	2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
7.	CRWMS M&O 1999c. *M&O Site Investigations*. Activity Evaluation. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990817.0257.	Entire	N/A - Reference only	2	Activity Evaluation	N/A	N/A	N/A	N/A
8.	CRWMS M&O 1999d. *Performance Assessment Operations*. Activity Evaluation. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990716.0106.	Entire	N/A - Reference only	2	Activity Evaluation	N/A	N/A	N/A	N/A
9.	CRWMS M&O 2000a. *Analysis of Base-Case Particle Tracking Results of the Base-Case Flow Fields*. ANL-NBS-HS-000024. Las Vegas, Nevada: CRWMS M&O.	p. 21–22	N/A - Reference only	6.4	Model Comparison	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
10. CRWMS M&O 2000b. *Particle Tracking Model and Abstraction of Transport Processes.* ANL-NBS-HS-000026. Las Vegas, Nevada: CRWMS M&O. URN-0037	6.2.4	N/A – Reference only	6.1	Description of Methodology	N/A	N/A	N/A	N/A
11. Doughty, C. 1999. "Investigation of Conceptual and Numerical Approaches for Evaluating Moisture, Gas, Chemical, and Heat Transport in Fractured Unsaturated Rock." *Journal Of Contaminant Hydrology*, *38* (1–3), 69–106. Amsterdam, The Netherlands: Elsevier Science Publishers. TIC: 244160.	pp. 100–104	N/A – Reference only	5.1	Rationale for dual continua approach	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
12. Dyer, J.R. 1999. "Revised Interim Guidance Pending Issuance of New U.S. Nuclear Regulatory Commission (NRC) Regulations (Revision 01, July 22, 1999), for Yucca Mountain, Nevada." Letter from J.R. Dyer (DOE) to D.R. Wilkins (CRWMS M&O), September 9, 1999, OL&RC:SB-1714, with enclosure, "Interim Guidance Pending Issuance of New U.S. Nuclear Regulatory Commission (NRC) Regulations (Revision 01)." ACC: MOL.19990910.0079.	Entire	N/A-Reference only	4.2	Interim guidance	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
13. LaBolle, E.M.; Fogg; G.E.; and Tompson, A. F.B. 1996. "Random-Walk Simulation of Transport in Heterogeneous Porous Media: Local Mass-Conservation Problem and Implementation Methods." *Water Resources Research*, 32 (4), 583–593. Washington, D.C.: American Geophysical Union. TIC: 245563.	pp. 583–593	N/A - Reference only	6.2.2	Equations 3 and 10	N/A	N/A	N/A	N/A
14. Pruess, K. 1991. *TOUGH2-A General Purpose Numerical Simulator for Multiphase Fluid and Heat Flow*. Report LBL-29400. Berkeley, California: Lawrence Berkeley National Laboratory. ACC: NNA.19940202.0088.	Entire	N/A - Reference only	6.3	General software use	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
15. Sudicky, E.A. and Frind, E.O. 1982. "Contaminant Transport in Fractured Porous Media: Analytical Solutions for a System of Parallel Fractures." *Water Resources Research, 18* (6), 1634–1642. Washington, D.C.: American Geophysical Union. TIC: 217475.	PP. 1634–1642	N/A - Reference only	6.4	Used analytical solution	N/A	N/A	N/A	N/A
16. Tang, D.H.; Frind, E.O; and Sudicky, E.A 1981. "Contaminant Transport in Fractured Porous Media: Analytical Solution for a Single Fracture." *Water Resources Research, 17* (3), 555–564. Washington, D.C.: American Geophysical Union. TIC: 225358.	pp. 555–564	N/A-Reference only	6.4.3	Analytical solution for matrix diffusion model.	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
17. Wu, Y.S.; Ahlers, C.F.; Fraser, P.; Simmons, A.; and Pruess, K. 1996. *Software Qualification of Selected TOUGH2 Modules*. Report LBNL-39490. Berkeley, California: Lawrence Berkeley National Laboratory. ACC: MOL.19970219.0104.	Entire	N/A – Reference only	6.3	General software use	N/A	N/A	N/A	N/A
18. Wu, Y. S. and K. Pruess. 1998. "A 3-D Hydrodynamic Dispersion Model for Modeling Tracer Transport in Geothermal Reservoirs." *Proceedings, Twenty-third Workshop, Geothermal Reservoir Engineering, Stanford, California, January 26–28, 1998*, 139–146. Stanford, California: Stanford University. TIC: 245292.	pp.139–146	N/A - Reference only	6.3	Projected area weighting method	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
19. Wu, Y.S.; Haukwa, C. and Bodvarsson, G.S. 1999. "A Site-Scale Model for Fluid and Heat Flow in the Unsaturated Zone of Yucca Mountain, Nevada." *Journal Of Contaminant Hydrology*, *38* (1–3), 185–215. Amsterdam, The Netherlands: Elsevier Science Publishers. TIC: 244160.	pp. 187–188 190–193	N/A – Reference only	5.1 5.3	Rationale for dual continua approach Discussion of approximations in numerical methods	N/A	N/A	N/A	N/A
20. Software Code: TOUGH2 V.1.4, STN: 10007-1.4-01.	Entire	N/A-Qualified/ Verified/ Confirmed	6	General software use	N/A	N/A	N/A	N/A
21. Software Code: T2R3D V.1.4, STN: 10006-1.4-00.	Entire	N/A-Qualified/ Verified/ Confirmed	6	General software use	N/A	N/A	N/A	N/A
22. Software Code: DCPT V.1.0, STN: 10078-1.0-00	Entire	TBV-3156	6	General software use	1	✓	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
23. Software Code: FEHM V.2.0, STN: 10031-2.00-00	Entire	N/A-Qualified/ Verified/ Confirmed	6	General software use	N/A	N/A	N/A	N/A
24. Macro/Routine: MAKEPTRK V.1.0. ACC: MOL.19990915.0361.	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to create transport parameter file for FEHM	N/A	N/A	N/A	N/A
25. Macro/Routine: PROCESS V.1.0. ACC: MOL.19990915.0360.	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to post-process results of FEHM particle-tracking simulations	N/A	N/A	N/A	N/A
26. Macro/Routine: T2FEHM2 V.2.0. ACC: MOL.19990915.0359	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to create FEHM-readable files from TOUGH2.	N/A	N/A	N/A	N/A
27. Macro/Routine: PrepareKDfile V1.0. ACC: MOL.20000127.0120.	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to create a DCPT-readable file from a TOUGH mesh file and a T2R3D input file	N/A	N/A	N/A	N/A
28. Macro/Routine: ExtractFlow V1.0. ACC: MOL.20000127.0121.	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to create a DCPT-readable file from the TOUGH output file.	N/A	N/A	N/A	N/A

OFFICE OF CIVILIAN RADIOACTIVE WASTE MANAGEMENT
DOCUMENT INPUT REFERENCE SHEET

1. Document Identifier No./Rev ANL-NBS-HS-000001/00		Change:	Title: Analysis Comparing Advective-Dispersive Transport Solution to Particle Tracking					
Input Document		4. Input Status	5. Section Used in	6. Input Description	7. TBV/TBD Priority	8. TBV Due To		
2. Technical Product Input Source Title and Identifier(s) with Version	3. Section					Unqual.	From Uncontrolled Source	Un-confirmed
29. Macro/Routine: ExBT V1.0. ACC: MOL.20000127.0122.	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to extract breakthrough curve from T2R3D output files.	N/A	N/A	N/A	N/A
30. Macro/Routine: StatSpatial V1.0. ACC: MOL.20000202.0193	Entire	N/A-Qualified/ Verified/ Confirmed	3	Routine to calculate the distribution of particles along y=10.25 based on the output file of DCPT.	N/A	N/A	N/A	N/A

AP-3.15Q.1 Rev. 06/30/1999

ATTACHMENT II—INPUT & OUTPUT FILES FOR DCPT AND FEHM

1. FILES FOR DCPT

All the files listed here will be submitted with this AMR. The typical steps used in simulation with DCPT (the technetium case as an example) are shown below:

Step 1: Prepare the input files and copy “UZ99.in” to “PTInput.txt”

Step 2: Execute ParticleTrack.exe

Step 3: Use standard spreadsheet software (Corel Quattro Pro 7.0), which is not subject to QARD, to calculate statistics of the exit time of particles contained in the file “UZ99_out.txt”, e.g., cumulative frequency scaled by the total number of particles.

Table II-1. Files Involved in Section 6.4.1

Filename	Description
“FM1DR.in”	Control file. List of all input/output files and parameters used by DCPT
"FM1D_m.tec"	List of the flow field and other transport parameters of the matrix for each cell
"FM1D_f.tec"	List of the flow field and other transport parameters of the fracture for each cell
“FM1D.txt”	Mesh file (cells, columns, and segments)
“FM1Dini.txt”	List of initial distribution of particles
“FM1DOutR.txt”	Output file, list of the final status of particles
“Fm1D.wb3”	A “Corel Quattro Pro 7” file which contains all post-process results and comparisons with the analytical solutions

Table II-2. Files Involved in Section 6.4.2

Filename	Description
“Analy3D.in”	Control file. List of all input/output files and parameters used by DCPT
“Analy3D_m.tec”	List of the flow field and other transport parameters of the matrix for each cell
“Analy3D_f.tec”	List of the flow field and other transport parameters of the fracture for each cell
“Analy3D.txt”	Mesh file (cells, columns, and segments)
“Ana3Dtextini.txt”	List of initial distribution of particles
“Analy3DOut.txt”	Temporary output file, list of the final status of particles
“ana3D2M.out”	Output file, distribution of particles along the specific line in space (y=0)
“Analy3d.wb3”	A “Corel Quattro Pro 7” file which contains all post-process results and comparisons with the analytical solutions

Table II-3. Files Involved in Section 6.4.3 (Comparison of DCPT Part Only)

Filename	Description
“UZ97_1D.in”	Control file. List of all input/output files and parameters used by DCPT for the case without sorption (Technetium)
“UZ97_1Dm.TEC”	List of the flow field and other transport parameters of the matrix for each cell
“Uz97_1df.tec”	List of the flow field and other transport parameters of the fracture for each cell
“UZ97_1DR.mesh”	Mesh file (cells, columns, and segments)
“UZ97_1DPT.ini”	List of initial distribution of particles
“UZ97_1Dout.txt”	Temporary output file, list of the final status of particles. The results are loaded into the spreadsheet file before another run of DCPT
“UZ97_1DFMD.dat”	List of the characteristic distances of the fracture systems in each cell
“UZ97_1D.flow”	List water flow rates (via both fracture and matrix) per connections of neighboring cells (part of TOUGH2 output)
“UZ97_1Dcon.dat”	Configuration of cell connections in the grid
“UZ97_1D.kd”	List of values of Kd and bulk density of related cells
“UZ97_1DR.in”	Control file. List of all input/output files and parameters used by DCPT for the case with sorption (Neptunium)
“UZ971D.wb3”	A “Corel Quattro Pro 7” file which contains all post-process results and comparisons with the numerical solutions for the case without sorption (Technetium)
“Uz971DR.wb3”	A “Corel Quattro Pro 7” file which contains all post-process results and comparisons with the numerical solutions for the case with sorption (Neptunium)

Table II-4. Files Involved in Section 6.4.4

Filename	Description
“UZ99.in”	Control file. List of all input/output files and parameters used by DCPT for the case without sorption (Technetium)
“UZ99_m.tec”	List of the flow field and other transport parameters of the matrix for each cell
“UZ99_f.tec”	List of the flow field and other transport parameters of the fracture for each cell
“UZ99mesh.txt”	Mesh file (cells, columns, and segments)
“UZ99PTini.txt”	List of initial distribution of particles
“UZ99_out.txt”	Output file, list of the final status of particles
“UZ99.flow”	List water flow rates (via both fracture and matrix) per connections of neighboring cells (part of TOUGH2 output)
“UZ99mesh.con”	Configuration of cell connections in the grid
“UZ99.kd”	List of values of Kd and bulk density of related cells
“UZ99DR.in”	Control file. List of all input/output files and parameters used by DCPT for the case with sorption (Neptunium)
“UZ99.wb3”	A “Corel Quattro Pro 7” file which contains all post-process results and comparisons with the numerical solutions for the case without sorption (Technetium)

2. FILES FOR FEHM

The software and files used in this analysis have been submitted to the Technical Data Management System (TDMS) as part of the records submittal of this analysis (DTN: SN9908T0581699.001). A complete explanation of the files is contained in README files in each directory. For the runs specific to the FEHM particle-tracking comparison, the files are contained in the tar file AMR_U0155_Ho.tar. The tar file may be zipped and contains a .gz suffix. Any decompression software (e.g., WinZip) should be able to decompress the files and un-tar (extract) the subdirectories. In Unix, type "gunzip AMR_U0155_Ho.tar" to unzip the file. Then, type "tar xvf AMR_U0155_Ho" to extract the subdirectories. The following provides a description of the files and how they are used in the development and implementation of the FEHM particle tracking simulations.

The 1-D TOUGH2 flow field is described by three files: sd9_e9.dt1, sd9_e9.ot1, and sd9_mesh. The rock properties and hydrologic properties are contained in sd9_e9.dt1 along with the infiltration source. The grid information is in sd9_mesh, and the output from the simulation is contained in sd9_e9.ot1. These files are used by T2FEHM2 to create FEHM-readable files that contain the same information. A complete description of the FEHM files created by T2FEHM2 can be found in Attachment III. The actual files are included and documented in the subdirectory 't2fehm2_files' in DTN: SN9908T0581699.001. T2FEHM2 is run only once since only one flow field is used in the com-parison study. The resulting files have the prefix 'fmsd9_e9.'

An additional file not created by T2FEHM2 is required for the FEHM particle-tracking simulations. The 'ptrk' macro file contains transport parameter information for different materials in the model and is created by MAKEPTRK (see Attachment III). This pre-processor uses the sd9_e9.dt1 and sd9_mesh files as input. In addition, it requires user-specified information on transport properties such as sorption coefficients, diffusion coefficients, and dispersivities.

ATTACHMENT III

SOFTWARE ROUTINES

Software Routine Name: **MAKEPTRK**
Version: 1.0
Development Software: FORTRAN 77, Sun OS 5.7

Description:

This is a software routine that creates the 'ptrk' macro in FEHM that describes the transport parameters for the particle tracking simulation. The 'ptrk' macro file contains transport parameter information for different materials in the model and is created by MAKEPTRK. This pre-processor uses the TOUGH2 ROCKS property file and mesh file as input to identify the different materials and the elements (nodes) that belong to those materials. In addition, it requires user-specified information on transport properties such as sorption coefficients, diffusion coefficients, and dispersivities. Below is a sample user-specified input file ("Np_diff.inp") for the Neptunium particle tracking simulation with diffusion and dispersion (an explanation of each line entry and the actual input parameter name from the source file is given following the dashed line):

```
sd9_e9.dt1
sd9_mesh
Np_diff.ptrk
1
4
1.
4.
0.
0.
20.
1.6e-10
1.
1
--------------------------------------
      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ROCKS card?'
      read(*,*) rocks
      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ELEME and CONNE cards?'
      read(*,*) mesh
      write(*,*)'What would you like to name the output file?'
      read(*,*) out
      write(*,*)'What transport mechanisms apply for the matrix?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff'
      read(*,*) iflagm
      write(*,*)'What transport mechanisms apply for the fracture?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff'
      read(*,*) iflagf
      write(*,*)'What is the Kd (cc/g) for vitric units?'
      read(*,*)xkdv
      write(*,*)'What is the Kd (cc/g) for zeolitic units?'
      read(*,*)xkdz
      write(*,*)'What is the Kd (cc/g) for devitrified units?'
      read(*,*)xkdd
```

```
      write(*,*)'What is the matrix dispersivity (m)?'
      read(*,*) dispm
      write(*,*)'What is the fracture dispersivity (m)?'
      read(*,*) dispf
      write(*,*)'What is the molecular diffusion coefficient?'
      read(*,*) do
      write(*,*)'What is the retardation factor for fracture'
      write(*,*)'sorption?  (1 = no fracture sorption)'
      read(*,*) rdfrac
      write(*,*)'Would you like to use the f/m reduction factor in'
      write(*,*)'calculating aperture parameter?  (1=yes,0=no)'
      read(*,*) nfm
```

Because this routine simply reads the input parameters and places them into a formatted output file, there is no limitation as to the range of input parameters that is used. The parameters can be visually inspected to ensure that the input values have been correctly transferred to the output file (see verification below). The output from MAKEPTRK is a file that contains transport parameter information in a format that is required by the FEHM 'ptrk' macro. The information is pasted into a 'master.ptrk' file and renamed. A sample of a resulting 'ptrk' file for Neptunium with diffusion, dispersion, and sorption ('fmNp_diff.ptrk') that is used by FEHM is provided below:

```
ptrk                    /* Np simulation with diffusion and dispersion
100000 204853          /* 100,000 particles, random # seed 204853 */
0 1.e20 0. 1.e20       /* time for starting, ending trans. simulation,and time for ending,
starting flow simultaion */
1 0 2 2                /* print out particle information and store it in *.fin */
1  0.000E+00  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.890E-01  0.100E-03 # 12 tswM4
1  0.000E+00  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.115E+00  0.100E-03 # 13 tswM5
1  0.000E+00  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.920E-01  0.100E-03 # 14 tswM6
1  0.000E+00  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.200E-01  0.100E-03 # 15 tswM7
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.265E+00  0.100E-03 # 16 ch1Mv
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.321E+00  0.100E-03 # 17 ch2Mv
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.321E+00  0.100E-03 # 18 ch3Mv
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.321E+00  0.100E-03 # 19 ch4Mv
1  0.400E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.193E+00  0.100E-03 # 20 ch1Mz
1  0.400E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.240E+00  0.100E-03 # 21 ch2Mz
1  0.400E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.240E+00  0.100E-03 # 22 ch3Mz
1  0.400E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.169E+00  0.100E-03 # 23 ch4Mz
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.274E+00  0.100E-03 # 24 pp3Mv
1  0.400E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.197E+00  0.100E-03 # 25 pp2Mz
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.274E+00  0.100E-03 # 26 bf3Mv
1  0.400E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.197E+00  0.100E-03 # 27 bf2Mz
1  0.100E+01  0.000E+00  0.000E+00  0.000E+00 0.160E-09  1.0  0.274E+00  0.100E-03 # 28 tm3Mv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.890E-01  0.412E-02 # 74 tswF4
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.115E+00  0.114E-01 # 75 tswF5
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.920E-01  0.119E-01 # 76 tswF6
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.200E-01  0.103E-01 # 77 tswF7
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.265E+00  0.930E-02 # 78 ch1Fv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.321E+00  0.100E-03 # 79 ch2Fv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.321E+00  0.100E-03 # 80 ch3Fv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.321E+00  0.100E-03 # 81 ch4Fv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.193E+00  0.821E-04 # 82 ch1Fz
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.240E+00  0.821E-04 # 83 ch2Fz
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.240E+00  0.821E-04 # 84 ch3Fz
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.169E+00  0.821E-04 # 85 ch4Fz
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.274E+00  0.100E-03 # 86 pp3Fv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.197E+00  0.100E-03 # 87 pp2Fz
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.274E+00  0.100E-03 # 88 bf3Fv
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.197E+00  0.100E-03 # 89 bf2Fz
4  0.000E+00  0.200E+02  0.200E+02  0.200E+02 0.160E-09  1.0  0.274E+00  0.100E-03 # 90 tm3Fv
1  1.00  0.000 0.00  0.00 0.160E-09  1.0  0.100E+00  0.100E-03 # 58 chaMd 35
4  0.00 20.00 20.00 20.00  0.160E-09  1.0  0.100E+00  0.164E-03 #115 chaFd 36
```

```
1 0 0 1
  -12 0 0    1
  -13 0 0    2
  -14 0 0    3
  -15 0 0    4
  -16 0 0    5
  -17 0 0    6
  -18 0 0    7
  -19 0 0    8
  -20 0 0    9
  -21 0 0    10
  -22 0 0    11
  -23 0 0    12
  -24 0 0    13
  -25 0 0    14
  -26 0 0    15
  -27 0 0    16
  -28 0 0    17
  -74 0 0    18
  -75 0 0    19
  -76 0 0    20
  -77 0 0    21
  -78 0 0    22
  -79 0 0    23
  -80 0 0    24
  -81 0 0    25
  -82 0 0    26
  -83 0 0    27
  -84 0 0    28
  -85 0 0    29
  -86 0 0    30
  -87 0 0    31
  -88 0 0    32
  -89 0 0    33
  -90 0 0    34
  -58 0 0    35
 -115 0 0    36

-500 0 0 -1. 0. 1.E-5    /* release particles at zone 500  */
```

The first four lines (note that the third line is wrapped) contain information for FEHM and are not relevant to the routine MAKEPTRK. The next 36 lines contain transport properties for different geologic layers of the system. These lines were extracted from the output of the MAKEPTRK file and only those materials at or beneath the repository were retained (geologic layers above the repository are not needed for simulations of radionuclide transport between the repository and the water table). The verification section below discusses the transport parameters in more detail. Following the blank line, the next 37 lines assign zones of nodes to each of the geologic layers. The final line is also irrelevant to MAKEPTRK and specifies the release of radionuclides.

Verification:

The sample output file shown above can be verified by visual inspection. A sample of a spot check is performed as follows. For material #74 (tswF4), the first column contains a flag that denotes the transport mechanism for this material. As identified in the input file, the transport mechanism for this fracture material should be denoted as “1” (advection, diffusion, and dispersion). The second column is the sorption coefficient, and it is correctly listed as “0” (for fractures). The next three columns are the dispersivity values for the x-, y-, and z-directions, and they are correctly listed as 20 m. The next column is the diffusion coefficient, which is correctly listed for Neptunium in the input file as 1.6E-10. The next column is the fracture sorption

parameter, which is correctly listed as "1." The next column is the corresponding matrix porosity (not actually used in this version of FEHM), which can be verified as correct by looking at the TOUGH2 ROCKS card (in 'sd9_e9.dt1' in DTN: SN9908T0581699.001) for material tswM4.

The last number to be verified in these rows of transport properties is the fracture aperture parameter that is used to simulate matrix diffusion. The aperture parameter is calculated as the fracture element volume divided by the fracture/matrix connection area for that fracture element. The fracture/matrix connection area can be found in the CONNE card (in 'sd9_mesh' in DTN: SN9908T0581699.001) for connections between fracture and matrix elements. In MAKEPTRK, the fracture/matrix connection area can also be calculated as the product of the connection area supplied in the CONNE card and the reduction factor, Xfm, found in the ROCKS card to accommodate reductions in fracture/matrix conductance due to sub-grid heterogeneities. The latter calculation was used for this analysis, but it was learned after these calculations were performed that the formulation in FEHM does not need a reduction in fracture/matrix area to be consistent with the prescribed flow fields (the fracture saturation, which represents this reduction factor, is already accommodated in the FEHM formulation for matrix diffusion). Future revisions of this analysis should revise the aperture parameter calculation to exclude the reduction factor, but the general trends and results are not expected to change significantly. The fracture volume for an element ('FlE71') belonging to the material 'tswF4' is given in the ELEME card (in 'sd9_mesh' in DTN: SN9908T0581699.001) as 355.8 m^3. The fracture/matrix connection area for this element is given in CONNE as 1.079E06 m^2. The reduction factor is given in the ROCKS card for 'tswF4' as 0.008. The aperture parameter (as used in this analysis) is therefore equal to (35.58 m^3) ÷ (1.079E06 m^2) ÷ (0.008) = 4.12E-3 m. This is exactly the value reported in the sample output file. Note that the aperture parameter is only relevant for fracture elements, so the values for the matrix elements are "dummy" parameters.

This verification ensures that MAKEPTRK is performing correctly for the range of input parameters that is used in this analysis.

Listing of Software Routine MAKEPTRK v. 1.0:

```
c  makeptrk_v1.f
c_____________________________________________________________________
c  This program will create the transport models that are used in the
c  FEHM ptrk macro.  The required input files are the TOUGH2 ROCKS card,
c  ELEME card, and CONNE card.  This program will also ask the user for
c  parameters including fracture and matrix diffusion, dispersivity, and
c  Kd.  The primary output is, for each ROCKS material, the Kd,
c  dispersivity, molecular diffusion, fracture sorption, matrix porosity,
c  and aperture parameter (for fracture->matrix diffusion).
c
c    C.K.Ho
c    3/12/99
c
c  Several modifications have been made:
c  1) Kd's are not assigned to fracture materials
c  2) Format for dispersivity value has been changed from f5.2 to e10.3
c  3) User is given an option to use fracture/matrix reduction factor in
```

```
c     calculating aperture parameter.
c    C.K.Ho
c    4/20/99
c_____________________________________________________________________________
c234567890123456789012345678901234567890123456789012345678901234567890123456789012
c     implicit double precision (a-h,o-z)
      character*22 block,rocks,mesh,out
      character*5 matname(999),mat(99999),elemn(99999),elem1,elem2
      real por(999),xfm(999),vf(99999),afm(99999),bf(999)

      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ROCKS card?'
      read(*,*) rocks
      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ELEME and CONNE cards?'
      read(*,*) mesh
      write(*,*)'What would you like to name the output file?'
      read(*,*) out
      write(*,*)'What transport mechanisms apply for the matrix?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff'
      read(*,*) iflagm
      write(*,*)'What transport mechanisms apply for the fracture?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff'
      read(*,*) iflagf
      write(*,*)'What is the Kd (cc/g) for vitric units?'
      read(*,*)xkdv
      write(*,*)'What is the Kd (cc/g) for zeolitic units?'
      read(*,*)xkdz
      write(*,*)'What is the Kd (cc/g) for devitrified units?'
      read(*,*)xkdd
      write(*,*)'What is the matrix dispersivity (m)?'
      read(*,*) dispm
      write(*,*)'What is the fracture dispersivity (m)?'
      read(*,*) dispf
      write(*,*)'What is the molecular diffusion coefficient?'
      read(*,*) do
      write(*,*)'What is the retardation factor for fracture'
      write(*,*)'sorption?  (1 = no fracture sorption)'
      read(*,*) rdfrac
      write(*,*)'Would you like to use the f/m reduction factor in'
      write(*,*)'calculating aperture parameter?  (1=yes,0=no)'
      read(*,*) nfm

      open(1,file=mesh,status='old')
      open(3,file=rocks,status='old')
      open(12,file=out,status='new')

c...Data
c...Assign a dummy aperture parameter for matrix materials.
c...Matrix diffusion is not used for matrix materials.
      bfm=1.e-4

c...Read in fracture information from MESH
      n=1
      read(1,1000) block
1000  format(a22)
99    read(1,65) elemn(n),mat(n),vf(n)
```

```
65    format(a5,10x,a5,e10.4)
c...End of active elements is signified by boundary elements or a
c...blank space
      if(elemn(n).eq.'     ') go to 98
      if(elemn(n)(1:2).eq.'TP'.or.
     & elemn(n)(1:2).eq.'BT') go to 99
      N=N+1
c...Read next line as matrix (assumes alternating listing)
      read(1,*)
      GO TO 99
98    CONTINUE
      NMAX = n - 1

c...Check
      do i=1,nmax
        write(*,*) vf(i),' ',mat(i)
      end do
c...End check

c...NMAX is the total number of fracture elements read from MESH
      write(*,107) nmax
107   format('Have read in ',i8,' fracture elements from MESH...')
c...nnodes is the total number of active nodes

c...Read in connection information from MESH
      N=1
c...Read header line CONNE
      READ(1,1500) BLOCK
1500  FORMAT(A22,3X,25X,E10.4)
c...Read elements 1 and 2 and the connection area for F/M pairs only
199   read(1,1502) elem1,elem2,afm(n)
1502  format(2a5,40x,e10.4)
      IF(elem1(1:5).EQ.'     '.OR.elem1(1:3).EQ.'+++') GO TO 198
      if(elem1(1:1).ne.'F'.or.elem2(1:1).ne.'M') go to 199
      N=N+1
      GO TO 199
198   CONTINUE
      NCMAX = N - 1
c...NCMAX is the total number of f/m connections read from MESH
      write(*,203) ncmax
203   format('Have read in ',i8,' f/m connections from MESH...')

c...Check
      do i=1,ncmax
        write(*,207) afm(i)
207     format(e10.4)
      end do
c...End check

c...Read in ROCKS information from TOUGH2 input file

18    read(3,1000) block
      if(block(1:5).ne.'ROCKS') go to 18

      i=1
      nfmat=0
      nmmat=0
408   read(3,410) matname(i),drok,por(i)
410   format(a5,5x,2e10.4)
      if(matname(i).eq.'REFCO') go to 408
      if(matname(i).eq.'     ') then
c...ntotmat is the total number of materials in the ROCKS card
        ntotmat=i-1
```

```
        go to 27
      end if
      read(3,*)
c...nfmat is the total number of fracture materials
      if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F') then
        nfmat=nfmat+1
      end if
c234567890123456789012345678901234567890123456789012345678901234567890123456789012
c...nmmat is the total number of matrix materials
      if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M') nmmat=nmmat+1
      read(3,33) xfm(i)
      if(xfm(i).eq.0.) xfm(i)=1.
33    format(60x,e10.4)
      read(3,*)
      i=i+1
      go to 408

27    continue
      write(*,75) nmmat
75    format('Number of matrix materials in ROCKS = ',i5)
      write(*,77) nfmat
77    format('Number of fracture materials in ROCKS = ',i5)

c...Check
      do i=1,ntotmat
        write(*,*) xfm(i)
      end do
c...End check

c...Determine matrix porosities corresponding to each fracture material
c...Because the number of fracture and matrix materials are not equal,
c...I am comparing the characters of the element names.  I first
c...determine where the 'F' is, and then I compare all other
c...characters with the matrix material to get a match.
      write(*,*) ntotmat
      do i=1,ntotmat
        if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M')goto83
        if(matname(i).eq.'topbd'.or.matname(i).eq.'botbd')goto83
          do j=1,ntotmat
            if(matname(i)(3:3).eq.'F') then
              if(matname(j)(3:3).eq.'M') then
                 if(matname(j)(1:2).eq.matname(i)(1:2).and.
     &              matname(j)(4:5).eq.matname(i)(4:5)) then
                    por(i)=por(j)
                    go to 83
                 end if
              end if
            elseif(matname(i)(4:4).eq.'F') then
              if(matname(j)(4:4).eq.'M') then
                 if(matname(j)(1:3).eq.matname(i)(1:3).and.
     &              matname(j)(5:5).eq.matname(i)(5:5)) then
                    por(i)=por(j)
                    go to 83
                 end if
              end if
            end if
          end do
          por(i)=0.1
c234567890123456789012345678901234567890123456789012345678901234567890123456789012
          write(*,113) matname(i)
113       format('Material ',a5,' does not have a matrix counterpart.'/
     &           'It has been assigned a matrix porosity of 0.1')
83    end do
```

```
c...Determine aperture parameter, bf

      do i=1,ntotmat

c...If material is boundary then assign a dummy aperture parameter
        if(matname(i).eq.'topbd'.or.matname(i).eq.'botbd') then
          bf(i)=bfm
          goto87
        end if

        if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F') then
          do j=1,nmax
            if(mat(j).eq.matname(i).and.nfm.eq.1) then
              bf(i)=vf(j)/(xfm(i)*afm(j))
c...Check
              write(*,*) bf(i)
              go to 87
            elseif(mat(j).eq.matname(i).and.nfm.eq.0) then
              bf(i)=vf(j)/afm(j)
c...End check
              go to 87
            end if
          end do

c...If a material cannot be associated with an active fracture element,
c...then assign the material a dummy aperture parameter.
          bf(i)=bfm

        end if
87    end do

c...Write data to output file for PTRK macro

      do i=1,ntotmat

c...Assign appropriate Kd
        if(matname(i)(5:5).eq.'v') then
          xkd=xkdv
        elseif(matname(i)(5:5).eq.'z') then
          xkd=xkdz
        else
          xkd=xkdd
        end if

c234567890123456789012345678901234567890123456789012345678901234567890123456789012
        if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M') then
          write(12,505)iflagm,xkd,dispm,dispm,dispm,do,1.,por(i),
     &                 bfm,i,matname(i)
505       format(i1,1x,4(e10.3,1x),e9.3,1x,f4.1,1x,2(e10.3,1x),'#',
     &           i3,1x,a5)
        else
          write(12,505)iflagf,0.,dispf,dispf,dispf,do,rdfrac,por(i),
     &                 bf(i),i,matname(i)
        end if
       end do

       write(12,*)

       do i=1,ntotmat
         write(12,507) -i,0,0,i
507      format(i5,1x,i1,1x,i1,1x,i5)
       end do
```

```
stop
end
```

Software Routine Name: **PROCESS1**
Version: 1.0
Development Software: Fujitsu FORTRAN 90

Description:

This is a software routine that post-processes the results of the FEHM particle tracking to provide columns of time, mass flux (mol/year) and cumulative mass at the water table (mol). The post-processor PROCESS1 is executed with an input file "process.dat" that is modified to reflect the desired output name of the run. This processor takes the information from the particle tracking code and prints the information to an output file named by the user. A sample input file ('process.dat') for PROCESS1 is given below:

```
../fmsd9_e9.grid
../fmsd9_e9.fin
fmNp_nodiff.output
0.5 100 100 1.000
4
0.01 0.05 0.1 0.5
```

The first two lines are the names of input files, and the third line is the desired output file name. The fourth line contains information about how the post-processor bins the particles for printing the time (years), mass flux (mol/years), and cumulative breakthrough (mol) at the water table. The fifth line indicates how many numbers are in the sixth line, and the sixth line contains values for the percent cumulative breakthrough at which times are desired to be printed to the screen.

The output file contains three columns. The first column is the time in years. The second column is the mass flow (mol/year) recorded at the water table. The third column is the cumulative mass (moles) that has reached the water table at the specified time. A sample of the output file 'fmNp_nodiff.output' is extracted below: The results of the PROCESS1 can be plotted directly.

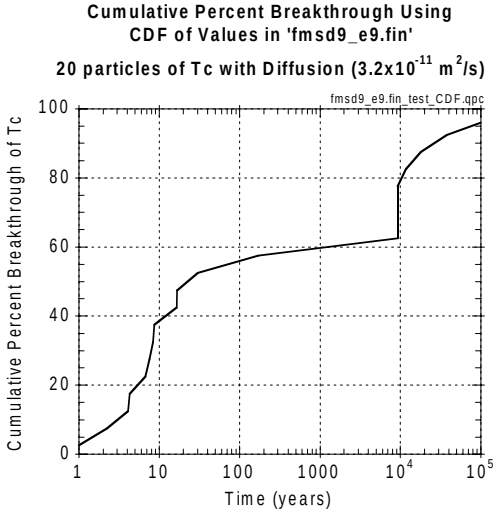
```
 1.16189623 9.70895290e-02 9.99999975e-06
 1.17165995 0.902203918 4.72000008e-03
 1.17587113 1.42056239 9.42999963e-03
 1.17873192 1.98488736 1.41399996e-02
 1.18089843 2.28967118 1.88500006e-02
 1.18277979 2.60619116 2.35600006e-02
 1.18448448 3.12997580 2.82700006e-02
 1.18586791 3.37901926 3.29799987e-02
 1.18721294 3.84510279 3.76899987e-02
 1.18845415 3.89304090 4.23999988e-02
 1.18956292 4.66002941 4.71099988e-02
 1.19055974 4.46301603 5.18199988e-02
 1.19154954 5.20216608 5.65299988e-02
 1.19247949 5.29256201 6.12399988e-02
     .
     .
     .
 366005.031 0.142051578 0.900529981
 366005.062 1.48951869e+10 0.905250013
```

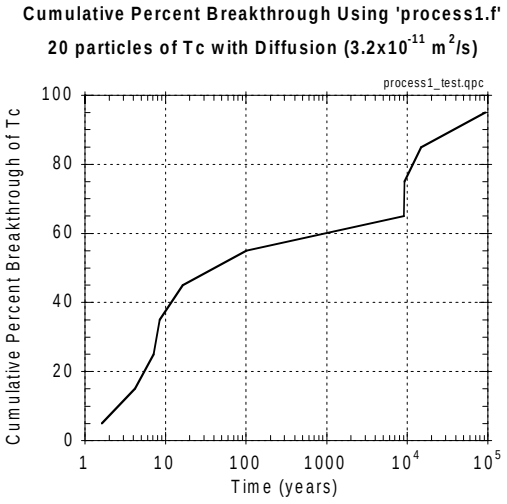
```
366005.062 0.142051578 0.909969985
366005.062 1.48951869e+10 0.914690018
366005.062 0.142051578 0.919409990
414162.375 3.62423052e-08 0.924130023
618860.563 3.03773398e-08 0.928849995
651621.313 1.48951869e+10 0.933570027
651621.313 7.10257888e-02 0.938290000
```

Verification:

This section contains a verification test of the software routine PROCESS1. The test consists of one of the 1-D simulations used in the FEHM particle tracking analysis (Tc with diffusion 3.2e-11 m^2/s and dispersion=20 m in fractures. Only 20 particles are used in the test case so that the output in the 'fmsd9_e9.fin' file can be directly processed and compared to the results of PROCESS1. The last row of numbers in the '*.fin' file contains the times at which each of the 20 particles left the system (exited at the water table). These times are plotted as a cumulative distribution function. The output from PROCESS1 is in 'fmTc_diff_test.output' (see /fehmruns/process/test_process in DTN: SN9908T0581699.001). The first column is the time in years. The second column is the interpolated mass flux (mol/yr), and the third column is the cumulative breakthrough (mol). Because only one mole was injected, the third column is the same as the cumulative percent breakthrough given in the CDF plotted directly from the '*.fin' file.

The following plots are reproduced from 'process1.doc' (see /fehmruns/process/test_process in DTN: SN9908T0581699.001), which show that the post-processor is producing results that are the same as the actual values in the output file. The post-processor provides interpolation, so that curve is smoothed in some regions. This verification indicates that PROCESS1 is performing correctly for the range of input parameters used in this analysis.





Listing of Software Routine PROCESS1 v. 1.0:

```
      program process1
      implicit none
      character(100) dummy_string, grid_file, fin_file, out_file
      character(4) gas_flag, ptrk_flag, dpdp_flag, dual_flag
      integer, allocatable :: ifinal(:), index(:)
      integer i, j, n0, npart, nseed, neq, ic, npartbin, nbins1, npbin
      integer nbins2, npstart, npartfin, nfraction_out, len_aaxy
      real(4) total_mass, flux, delta_time, sumtime, fraction1, fluxmax
      real(4) timemax
      real(8), allocatable :: rdum(:), a_axy(:)
      real(4), allocatable :: timep(:)
      real(4), allocatable :: fraction_out(:)
      integer, allocatable :: ifraction_out(:)

c     process.dat contains files names, histogram parameters

      open(1,file='process.dat')

c     Read name of grid file, .fin file, then open them


      read(1,'(a100)') grid_file
      read(1,'(a100)') fin_file
      read(1,'(a100)') out_file

      open(3,file= grid_file)
      open(4,file = fin_file)

c     Read number of nodes from grid file, then close

      read(3,'(a100)') dummy_string
      read(3,*) neq
      close(3)

c     open output file

      open(7,file= out_file)

c     Read 3 dummy lines, then get gas flag, ptrk flag

      read(4,'(a100)') dummy_string
      read(4,'(a100)') dummy_string
      read(4,'(a100)') dummy_string

      read(4,'(a4)') gas_flag
      read(4,'(a4)') ptrk_flag

c     Read dual and dpdp flags to tell if either option was used

      read(4,'(a100)') dummy_string
      read(4,'(a4)') dpdp_flag
      read(4,'(a4)') dual_flag

c     Set n0 based on ECM, DPDP, or DUAL

      if(dpdp_flag .eq. 'dpdp') then
         n0 = 2*neq
      elseif(dual_flag .eq. 'dual') then
```

```
         n0 = 3*neq
      else
         n0 = neq
      end if

c     Allocate space for the array to read state variables

      allocate(rdum(n0))

c     read in state variables based on what type of gas option was used

      if(gas_flag .eq. 'ngas') then
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
      elseif(gas_flag(1:3) .eq. 'air') then
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
      else
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
         read(4,'(4g20.10)')(rdum(i),i=1,n0)
      end if

c     rdum no longer needed

      deallocate(rdum)

c     Determine if mass fluxes are in file, read if they are

      read(4,'(a100)') dummy_string
      if(dummy_string(1:4).eq.'mass') then
         read(4,*) len_aaxy
         allocate(a_axy(len_aaxy))
         read(4,'(5g15.8)') (a_axy(i),i=1,len_aaxy)
         deallocate(a_axy)
      else
         backspace 4
      end if


c     Read in number of particles, seed value

      read(4,*) npart, nseed

c     Allocate space for final node array and time array

      allocate(ifinal(npart))
      allocate(index(npart))
      allocate(timep(npart))


      read(4,*)(ifinal(i),i=1,npart)

c     Skip through two other output arrays to get to the time array
c     if the user wrote these arrays out

      if(nseed .gt. 0) then
         read(4,*)(timep(i),i=1,npart)
         read(4,*)(timep(i),i=1,npart)
      end if
```

```
c     Read array of leaving times (or particle age if ifinal>0)

      read(4,*)(timep(i),i=1,npart)


c     Loop takes all particles that have left the system and shifts them
c     to the first positions in the arrays so that the sorting routine
c     will not have to deal with particles that are still in the system

      ic = 0
      do i = 1, npart

         if(ifinal(i) .lt. 0) then
            ic = ic + 1
            ifinal(ic) = ifinal(i)
            timep(ic) = timep(i)
         end if

      end do

c     The number of particles to bin are now only the ones that left
c     the system


      npartbin = ic

c     read in number of bins, total mass of radionuclides
c     fraction1 - the first set of bins applies to the first
c     fraction1*npartbin particles
c     nbins1 - number of bins in which to bin the first set of particles
c     nbins2 - number of bins for the remaining particles
c
      read(1,*) fraction1, nbins1, nbins2, total_mass
      read(1,*) nfraction_out
      allocate(fraction_out(nfraction_out))
      allocate(ifraction_out(nfraction_out))
      read(1,*)(fraction_out(i),i=1,nfraction_out)
      do i = 1, nfraction_out
         ifraction_out(i)=fraction_out(i)*npart
      end do

c     Call routine to sort the particles and node array
c     lowest to highest
c     This routine is an indexing and ranking sort routine that returns
c     the index array, such that timep(index(i)), i = 1, npartbin
c     are in ascending order. It doesn't sort the timep array itself,
c     but supplies the index array so that the order can be obtained
c     by indirect indexing

      call sort_parti(npartbin, npart, index, timep)

c     Set max flux to small number initially

      fluxmax = 0.

c     Do average time and mass flux calculation for each bin of first
c     partition

      npartfin = fraction1*npartbin
      npbin = npartfin/nbins1
      bin_loop1: do i = 1, npartfin, npbin
         sumtime = 0.
         if(i+npbin-1.gt.npartfin) exit bin_loop1
```

```
         do j = i, i+npbin-1
            sumtime = sumtime + timep(index(j))
         end do
         sumtime = sumtime / npbin
         delta_time = timep(index(i+npbin-1))-timep(index(i))
         if(delta_time.eq.0.) delta_time = 1.e-5
         flux = npbin * total_mass / (npart*delta_time)
         if(flux.gt.fluxmax) then
            fluxmax = flux
            timemax = sumtime/31557600.
         end if
         write(7,*) sumtime/31557600., 31557600.*flux,
     2        real(i)/real(npart)
      end do bin_loop1

c     Do average time and mass flux calculation for each bin of second
c     partition

      npstart = i
      npartfin = npartbin
      npbin = (npartbin-i+1)/nbins2
      bin_loop2: do i = npstart, npartfin, npbin
         sumtime = 0.
         if(i+npbin-1.gt.npartfin) exit bin_loop2
         do j = i, i+npbin-1
            sumtime = sumtime + timep(index(j))
         end do
         sumtime = sumtime / npbin
         delta_time = timep(index(i+npbin-1))-timep(index(i))
         if(delta_time.eq.0.) delta_time = 1.e-5
         flux = npbin * total_mass / (npart*delta_time)
         if(flux.gt.fluxmax) then
            fluxmax = flux
            timemax = sumtime/31557600.
         end if
         write(7,*) sumtime/31557600., 31557600.*flux,
     2        real(i)/real(npart)
      end do bin_loop2
      write(6,*)(timep(index(ifraction_out(i)))/31557600.,
     2     i=1,nfraction_out), timemax, 31557600.*fluxmax

      end

      subroutine sort_parti(n, nsize, indx, time)
c
c     Indexing and Ranking algorithm for sorting, from Numerical Recipes
c     Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T.
c     Vetterling, 1986, Numerical Recipes. The Art of Scientific
c     Computing, Cambridge University Press, Cambridge, pp. 232-234.
c
      implicit none
      real(4) time(nsize), q
      integer indx(nsize)
      integer nsize
      integer n, j, l, ir, indxt, i
      do j = 1, n
         indx(j) = j
      end do
      l = n/2+1
      ir=n
 10   continue
      if(l.gt.1) then
         l=l-1
```

```
         indxt=indx(l)
         q=time(indxt)
      else
         indxt=indx(ir)
         q=time(indxt)
         indx(ir)=indx(1)
         ir=ir-1
         if(ir.eq.1) then
            indx(1)=indxt
            return
         end if
      end if
      i=l
      j=l+l
20    if(j.le.ir) then
         if(j.lt.ir) then
            if(time(indx(j)).lt.time(indx(j+1))) j=j+1
         end if
         if(q.lt.time(indx(j))) then
            indx(i)=indx(j)
            i=j
            j=j+j
         else
            j=ir+1
         end if
         goto 20
      end if
      indx(i)=indxt
      goto 10
      end
```

Software Routine Name: **T2FEHM2**
Version: 2.0
Development Software: FORTRAN 77, Sun OS 5.7

Description:

The software routine T2FEHM2 was written to reformat TOUGH2 files that contain information pertaining to unsaturated flow to FEHM-readable files that can be used for radionuclide particle tracking. This method maintains consistency with the hydrologic conditions (mass flow rates, liquid saturations, etc.) prescribed in the TOUGH2 flow fields.

FEHM uses a cell-based particle tracking model that preserves the overall residence time through any portion of the model and probabilistically reproduces the migration of a solute through the domain. The requirement for the method is that the flow calculation be based on a control volume in which fluid flow rates into and out of each cell are computed. Since TOUGH2 is an integrated finite difference code, and FEHM employs a control volume finite element technique, the two codes are compatible for implementing the particle tracking technique. The required inputs for FEHM to use an externally-developed flow field are: (1) grid connectivity information and cell volumes; (2) properties and state variables (rock grain density, fluid saturation, and rock porosity at each grid point); (3) inter-nodal fluid mass flow rate for every connection in the numerical grid; and (4) fluid source and sink flow rates for each grid block. The post-processor, T2FEHM2, was written to generate these required data from existing TOUGH2 files. The remainder of this section describes the required inputs to T2FEHM2 and the corresponding output files.

Required Input Files for T2FEHM2

When executed, T2FEHM2 will prompt the user for the names of three required files: (1) TOUGH2 input file; (2) TOUGH2 output file; and (3) TOUGH2 mesh file. T2FEHM2 will also prompt the user for the name of a fourth file containing the names of repository elements, but this file is optional. All input files can be found in the subdirectory ‘fromLBNL’ in DTN: SN9908T0581699.001.

TOUGH2 Input File

The TOUGH2 input file must contain the ROCKS and GENER cards. ROCKS contains material property information for fracture and matrix materials corresponding to a dual-permeability model. Fracture and matrix materials must have an ‘F’ or ‘M’, respectively, in the third or fourth character of the material name. Each material must have four lines associated with its entry. The GENER card should contain information on the infiltration source terms for prescribed elements. The generation rate is specified in units of kg/s.

TOUGH2 Output File

The TOUGH2 output file contains all simulated state variables (pressure, saturation) for each element and flux variables (mass flow rate) for each connection pair at user-specified print-out

times. T2FEHM2 reads in these state and flux variables and puts them in a format that is compatible with FEHM.

TOUGH2 Mesh File

The TOUGH2 mesh file contains the ELEME and CONNE cards. ELEME contains the element names, material names, volumes, and coordinates of each element in the TOUGH2 model. The fracture and matrix elements should be listed alternately with a fracture element listed first. Also, all boundary elements must be listed at the end of the ELEME card. The material names associated with each element should be five-character names (not integers) that correspond identically to the name of one of the materials in the ROCKS card. The CONNE card contains all connection pairs and associated connection information for each element in the TOUGH2 model. T2FEHM2 stores these connection pairs to create connectivity arrays (ncon, istrw, nelmdg) for FEHM.

File Containing Repository Elements

A file containing the names of repository elements is optional. If present, T2FEHM2 will read the number of repository elements in the first line of the file. All repository element names will then be read from the file. These elements will be used to create special fracture and matrix zones in a FEHM file that will be used to define the location of radionuclide release for particle tracking. For the 1-D FEHM simulation used in this analysis, only one element is specified for the repository zone.

6.1.2 Output Files from T2FEHM2

After reading the required information from the input files, T2FEHM2 prints out nine (9) files that are used by FEHM. The user specifies a reference file name, and the code creates nine output files by appending the following nine suffixes to the reference file name:

.dat
.dpdp
.files
.grid
.ini
.rock
.stor
.zone
.zone2

All T2FEHM2 output files can be found in the subdirectory ‘t2fehm2_files’ in DTN: SN9908T0581699.001. A tenth file, ‘file_name.check,’ is also printed but it is not used by FEHM. This file contains the node numbers and number of connections for each node. More detailed information on the contents of the FEHM macros can be found in Zyvoloski et al. (1997). The user should consult this information because a number of these macros have been created with T2FEHM2 using “dummy” variables that are either not needed by the particle tracking simulation (e.g., permeability, area coefficients, element specifications for nodes, etc.)

or that can be modified by the user to suit the specific needs of the particle tracking simulation (e.g., date, time steps, print-out options, etc.). Most of these prescribed variables appear in the '*.dat' file, so the user should become familiar with the macros listed in that file before using the default values prescribed in T2FEHM2.

The prefix "fm" is placed in front of all T2FEHM2 files for identification purposes. The remainder of this section details the specific output files. To verify that T2FEHM2 is producing correct results, portions of the actual output files used in this analysis ('fmsd9_e9*') are included. The values are compared to those in the original TOUGH2 files by visual inspection to ensure correct results for the range of inputs used in this analysis.

Output File '*.dat'

This file contains the required macros used by FEHM: 'dpdp,' 'perm,' 'rlp,' 'rock,' 'flow,' 'time,' 'ctrl,' 'iter,' 'sol,' 'rflo,' 'air,' 'node,' 'zone,' 'ptrk.' If the macros are not explicitly defined in this file, the names of macro files containing the actual information are listed here. Macros 'perm' and 'rlp' are not required by the particle tracking solution, so dummy values are inserted here. In addition, many of the values in the '*.dat' file are prescribed within T2FEHM2 as default values, so the user should refer to Zyvoloski et al. (1997) to modify the values in the different macros to suit their needs. The output file 'fmsd9_e9.dat' is provided below :

```
"fmsd9_e9.dat" 47 lines, 770 characters
 # input file for mean alpha, fitted fmx, present day q, ysw # AR 11/19/97
# Particle tracking for TOUGH2 flow field
dpdp
file
fmsd9_e9.dpdp
perm
1  0  0   0.100E-14 0.100E-14 0.100E-14

rlp
1  0.  0.  1.  1.  0.  1.

1  0  0  1

rock
file
fmsd9_e9.rock
flow

time
  0.36525E+09  0.36525E+09      10      10    1997      10

ctrl
     -10  0.10E-03      40
       1       0       0       1
0
      1.00      3.00      1.00
       5  0.20E+01  0.10E-09  0.10E+11
       0       1
iter
  0.10E-04  0.10E-04  0.10E-04 -0.10E-03  0.12E+01
       0       0       0       0  0.14E+05
sol
       1      -1
rflo
```

```
air
-1
20.0  0.1
node
1
1
zone
file
fmsd9_e9.zone2
ptrk
file
fmsd9_e9.ptrk
stop
```

Output File '*.dpdp'

This file contains a list of the zones corresponding to the fracture materials and lists the fracture porosities. It also contains dummy information regarding the length scale for matrix nodes that is not required for the TOUGH2-FEHM coupling. Here are the first few lines from 'fmsd9_e9.dpdp' that can be compared to the ROCKS card used by TOUGH2:

```
dpdp
1
     -63       0       0     0.2330E-03
     -64       0       0     0.2990E-03
     -65       0       0     0.7050E-04
     -66       0       0     0.4840E-04
     -67       0       0     0.4830E-04
     -68       0       0     0.1300E-03
     -69       0       0     0.6940E-04
     -70       0       0     0.3860E-04
     -71       0       0     0.8920E-04
     -72       0       0     0.1290E-03
     -73       0       0     0.1050E-03
     -74       0       0     0.1240E-03
     -75       0       0     0.3290E-03
     -76       0       0     0.3990E-03
```

Output File '*.files'

This control file contains a list of files that FEHM reads for necessary information. Below is the 'fmsd9_e9.files' file:

```
fmsd9_e9.dat
fmsd9_e9.grid
fmsd9_e9.zone
fmsd9_e9.out
fmsd9_e9.ini
fmsd9_e9.fin
fmsd9_e9.his
fmsd9_e9.trc
fmsd9_e9.con

fmsd9_e9.stor
fmsd9_e9.chk
```

```
all
0
```

Output File '*.grid'

This file contains the 'coor' and 'elem' macros. The first line of the 'coor' macro gives the total number of fracture elements, followed by a list of all the nodes in the fracture domain and their respective x, y, and z coordinates. The 'elem' macro contains dummy information regarding the nodes associated with each element, but this is not required for the TOUGH2-FEHM coupling. Below are the first few lines of the 'fmsd9_e9.grid' file that can be compared to the values in ELEME used by TOUGH2:

```
coor
      25
       1    171270.58    234054.36      1289.80
       2    171270.58    234054.36      1285.89
       3    171270.58    234054.36      1281.98
       4    171270.58    234054.36      1275.11
       5    171270.58    234054.36      1263.82
       6    171270.58    234054.36      1255.06
       7    171270.58    234054.36      1242.07
       8    171270.58    234054.36      1224.77
       9    171270.58    234054.36      1214.57
```

Output File '*.ini'

This file contains re-start information for FEHM. The liquid saturations of all fracture and matrix nodes are listed following eight header lines. The gas-phase pressures (MPa) are then listed for the fracture and matrix nodes. The fourth header line ('air') tells FEHM that the pressures are for the gas phase. Then, mass flux values (kg/s) are listed for each connection of each node, starting with node 1 (the ordering is the same as the 'ncon' array in '.stor' without pointer information—see '*.stor' below). The mass flux values include sources (infiltration) denoted as negative values and sinks (connection to water table) denoted as positive values for each node. Flow into a node is negative, and flow out of a node is positive. The mass flux values for the fracture domain are listed first followed by the mass flux values in the matrix domain. The mass flux between fracture and matrix elements are listed last. Flow from the fracture to the matrix is denoted as positive. The file 'fmsd9_e9.ini' is shown below:

```
"fmsd9_e9.ini" 71 lines, 4506 characters
 # input file for mean alpha, fitted fmx, present day q, ysw # AR 11/19/97
This is a .ini file with saturations, pressures and mass flux values.
0.
air
ptrk
nstr
dpdp
ndua
  0.70676000E-01  0.76186000E-01  0.10618000      0.66245000E-01
  0.61774000E-01  0.37347000E-01  0.23308000E-01  0.22572000
  0.13596000      0.78086000E-01  0.69016000E-01  0.89885000E-01
  0.10026000      0.10023000      0.10020000      0.10021000
```

```
  0.12480000      0.12220000      0.17218000      0.26035000
  0.30390000      0.26406000      0.78540000      0.32361000
  0.28965000      0.95558000      0.95624000      0.99071000
  0.61900000      0.57734000      0.54192000      0.43204000
  0.41860000      0.76104000      0.56915000      0.80942000
  0.91995000      0.85113000      0.85505000      0.85825000
  0.85443000      0.96344000      0.59510000      0.74785000
  0.98919000      0.99920000      0.99482000      0.94023000
  0.96382000      0.99339000
  0.91999800E-01  0.92000100E-01  0.92000000E-01  0.92000000E-01
  0.92000000E-01  0.92000000E-01  0.92002000E-01  0.92000000E-01
  0.92000000E-01  0.92000000E-01  0.92000000E-01  0.92000000E-01
  0.91999600E-01  0.91999800E-01  0.92000100E-01  0.91999800E-01
  0.91999500E-01  0.92000400E-01  0.91999600E-01  0.91999700E-01
  0.92000300E-01  0.92000000E-01  0.92000000E-01  0.92000400E-01
  0.91999900E-01  0.92000000E-01  0.92000000E-01  0.92000000E-01
  0.91995000E-01  0.92000000E-01  0.92004000E-01  0.92004000E-01
  0.92000000E-01  0.92000000E-01  0.92000000E-01  0.92000000E-01
  0.92000000E-01  0.92000000E-01  0.92000000E-01  0.92000000E-01
  0.92000000E-01  0.92000000E-01  0.91998000E-01  0.92000000E-01
  0.92000000E-01  0.91999800E-01  0.92000000E-01  0.92000000E-01
  0.91998000E-01  0.92005000E-01
mass flux values
     171     #ntotmfv=     146, nnodes=      50, number of f-m connections=       25
-0.12390000E-02 0.12390000E-02-0.12390000E-02         0.     0.12390000E-02
-0.12390000E-02         0.     0.12387000E-02-0.12387000E-02         0.
 0.20896000E-04-0.20896000E-04         0.     0.59733000E-05-0.59733000E-05
         0.     0.61827000E-06-0.61827000E-06         0.     0.20613000E-08
-0.20613000E-08         0.     0.98473000E-05-0.98473000E-05         0.
 0.12246000E-02-0.12246000E-02         0.     0.12229000E-02-0.12229000E-02
         0.     0.12229000E-02-0.12229000E-02         0.     0.12224000E-02
-0.12224000E-02         0.     0.12211000E-02-0.12211000E-02         0.
 0.12196000E-02-0.12196000E-02         0.     0.12175000E-02-0.12175000E-02
         0.     0.12159000E-02-0.12159000E-02         0.     0.11714000E-02
-0.11714000E-02         0.     0.10451000E-02-0.10451000E-02         0.
 0.83611000E-03-0.83611000E-03         0.     0.83487000E-03-0.83487000E-03
         0.     0.86962000E-03-0.86962000E-03         0.     0.86860000E-03
-0.86860000E-03         0.     0.11925000E-02-0.11925000E-02         0.
 0.11749000E-02-0.11749000E-02 0.11520000E-02         0.     0.60741000E-08
-0.60741000E-08         0.     0.31222000E-07-0.31222000E-07         0.
 0.33953000E-06-0.33953000E-06         0.     0.12181000E-02-0.12181000E-02
         0.     0.12330000E-02-0.12330000E-02         0.     0.12384000E-02
-0.12384000E-02         0.     0.12390000E-02-0.12390000E-02         0.
 0.12292000E-02-0.12292000E-02         0.     0.14363000E-04-0.14363000E-04
         0.     0.16063000E-04-0.16063000E-04         0.     0.16102000E-04
-0.16102000E-04         0.     0.16639000E-04-0.16639000E-04         0.
 0.17937000E-04-0.17937000E-04         0.     0.19370000E-04-0.19370000E-04
         0.     0.21459000E-04-0.21459000E-04         0.     0.23079000E-04
-0.23079000E-04         0.     0.67622000E-04-0.67622000E-04         0.
 0.19386000E-03-0.19386000E-03         0.     0.40289000E-03-0.40289000E-03
         0.     0.40413000E-03-0.40413000E-03         0.     0.36938000E-03
-0.36938000E-03         0.     0.37040000E-03-0.37040000E-03         0.
 0.46502000E-04-0.46502000E-04         0.     0.64082000E-04-0.64082000E-04
 0.86966000E-04 0.60741000E-08 0.25148000E-07 0.30831000E-06 0.12178000E-02
 0.14922000E-04 0.53551000E-05 0.61621000E-06-0.98452000E-05-0.12148000E-02
 0.16997000E-05 0.39210000E-07 0.53745000E-06 0.12978000E-05 0.14333000E-05
 0.20891000E-05 0.16192000E-05 0.44544000E-04 0.12624000E-03 0.20903000E-03
 0.12428000E-05-0.34750000E-04 0.10209000E-05-0.32390000E-03 0.17580000E-04
 0.22884000E-04
```

This file can be verified by comparing the saturations and mass fluxes to the actual values reported in the TOUGH2 output file ('sd9_e9.ot1' in DTN: SN9908T0581699.001). The first fracture element listed ('FaE71') is used to spot check these values. In 'sd9_e9.ot1' the liquid saturation is reported to be 0.70676E-01, which corresponds exactly to the saturation reported for the first fracture element in 'fmsd9_e9.ini' above. The mass flow rate between 'FaE71' and the second element below it ('FbE71') is reported in the TOUGH2 output file in CONNE as 0.12390E-02 kg/s. In the 'fmsd9_e9.ini' file this value can be found under the heading 'mass flux values' beneath the first header line. This value is actually the second number listed. The first number, which is identical in value but negative, represents the generation of mass flow originating from infiltration in the upper boundary element. The mass flow between the fracture and matrix elements corresponding to 'FaE71' can also be verified. In the TOUGH2 output file, the mass flow between 'FaE71' and 'MaE71' is given in CONNE as -0.60741E-08 kg/s (which denotes flow from the fracture to the matrix. The corresponding value can be found in 'fmsd9_e9.ini' by noting that there are 25 active nodes in each continuum. Therefore, this value should be the 25th value from the last number in the file. A visual check in 'fmsd9_e9.ini' shown above confirms that this value is correctly listed.

Output File '*.rock'

This file lists the zones of all fracture and matrix materials. For each zone, the rock grain density (kg/m^3), specific heat (J/kg-K), matrix porosity, and intrinsic fracture porosity (1) are listed. The output file 'fmsd9_e9.rock' is shown below, and values can be confirmed with the values in the ROCK card of 'sd9_e9.dt1.'

```
"fmsd9_e9.rock" 124 lines, 8481 characters
rock
      -1       0       0     0.2480E+04     0.1000E+04     0.6600E-01
      -2       0       0     0.2480E+04     0.1000E+04     0.6600E-01
      -3       0       0     0.2480E+04     0.1000E+04     0.1400E+00
      -4       0       0     0.2300E+04     0.1000E+04     0.3690E+00
      -5       0       0     0.2300E+04     0.1000E+04     0.2340E+00
      -6       0       0     0.2300E+04     0.1000E+04     0.3530E+00
      -7       0       0     0.2300E+04     0.1000E+04     0.4690E+00
      -8       0       0     0.2300E+04     0.1000E+04     0.4640E+00
      -9       0       0     0.2480E+04     0.1000E+04     0.4200E-01
     -10       0       0     0.2480E+04     0.1000E+04     0.1460E+00
     -11       0       0     0.2480E+04     0.1000E+04     0.1350E+00
     -12       0       0     0.2480E+04     0.1000E+04     0.8900E-01
     -13       0       0     0.2480E+04     0.1000E+04     0.1150E+00
     -14       0       0     0.2480E+04     0.1000E+04     0.9200E-01
     -15       0       0     0.2480E+04     0.1000E+04     0.2000E-01
     -16       0       0     0.2300E+04     0.1000E+04     0.2650E+00
     -17       0       0     0.2300E+04     0.1000E+04     0.3210E+00
     -18       0       0     0.2300E+04     0.1000E+04     0.3210E+00
     -19       0       0     0.2300E+04     0.1000E+04     0.3210E+00
     -20       0       0     0.2300E+04     0.1000E+04     0.1930E+00
     -21       0       0     0.2300E+04     0.1000E+04     0.2400E+00
     -22       0       0     0.2300E+04     0.1000E+04     0.2400E+00
     -23       0       0     0.2300E+04     0.1000E+04     0.1690E+00
     -24       0       0     0.2300E+04     0.1000E+04     0.2740E+00
     -25       0       0     0.2300E+04     0.1000E+04     0.1970E+00
     -26       0       0     0.2300E+04     0.1000E+04     0.2740E+00
```

-27	0	0	0.2300E+04	0.1000E+04	0.1970E+00
-28	0	0	0.2300E+04	0.1000E+04	0.2740E+00
-29	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-30	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-31	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-32	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-33	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-34	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-35	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-36	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-37	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-38	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-39	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-40	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-41	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-42	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-43	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-44	0	0	0.2390E+04	0.1000E+04	0.2000E+00
-45	0	0	0.2300E+04	0.1000E+04	0.2650E+00
-46	0	0	0.2300E+04	0.1000E+04	0.3210E+00
-47	0	0	0.2300E+04	0.1000E+04	0.2740E+00
-48	0	0	0.2300E+04	0.1000E+04	0.2650E+00
-49	0	0	0.2300E+04	0.1000E+04	0.3210E+00
-50	0	0	0.2300E+04	0.1000E+04	0.2740E+00
-51	0	0	0.2480E+04	0.1000E+04	0.3600E-01
-52	0	0	0.2300E+04	0.1000E+04	0.2880E+00
-53	0	0	0.2300E+04	0.1000E+04	0.2880E+00
-54	0	0	0.2300E+04	0.1000E+04	0.3320E+00
-55	0	0	0.2300E+04	0.1000E+04	0.3320E+00
-56	0	0	0.2300E+04	0.1000E+04	0.3320E+00
-57	0	0	0.2300E+04	0.1000E+04	0.2660E+00
-58	0	0	0.2300E+04	0.1000E+04	0.1000E+00
-59	0	0	0.2390E+04	0.1000E+04	0.5000E-01
-60	0	0	0.2390E+04	0.1000E+04	0.5000E-01
-61	0	0	0.2390E+04	0.1000E+04	0.5000E-01
-62	0	0	0.2390E+04	0.1000E+04	0.1000E+00
-63	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-64	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-65	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-66	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-67	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-68	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-69	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-70	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-71	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-72	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-73	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-74	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-75	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-76	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-77	0	0	0.2480E+04	0.1000E+04	0.1000E+01
-78	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-79	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-80	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-81	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-82	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-83	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-84	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-85	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-86	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-87	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-88	0	0	0.2300E+04	0.1000E+04	0.1000E+01
-89	0	0	0.2300E+04	0.1000E+04	0.1000E+01

```
     -90       0       0     0.2300E+04     0.1000E+04     0.1000E+01
     -91       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -92       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -93       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -94       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -95       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -96       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -97       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -98       0       0     0.2390E+04     0.1000E+04     0.1000E+01
     -99       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -100       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -101       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -102       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -103       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -104       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -105       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -106       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -107       0       0     0.2480E+04     0.1000E+04     0.1000E+01
    -108       0       0     0.2480E+04     0.1000E+04     0.1000E+01
    -109       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -110       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -111       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -112       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -113       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -114       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -115       0       0     0.2300E+04     0.1000E+04     0.1000E+01
    -116       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -117       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -118       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -119       0       0     0.2390E+04     0.1000E+04     0.1000E+01
    -120       0       0     0.2300E+04     0.1000E+04     0.2000E+00
    -121       0       0     0.2300E+04     0.1000E+04     0.2800E+00

stop
```

Output File '*.stor'

The file contains connectivity arrays and control volumes for the grid. Following two header lines, four integers are listed:

iwtotl: Total number of connections in a continuum (either fracture or matrix) for which inter-node fluxes and areas are assigned. This includes connections for a node to itself for sources and sinks. Equal to ncont-(neq+1).

neq: Number of nodes in either the fracture or matrix continuum.

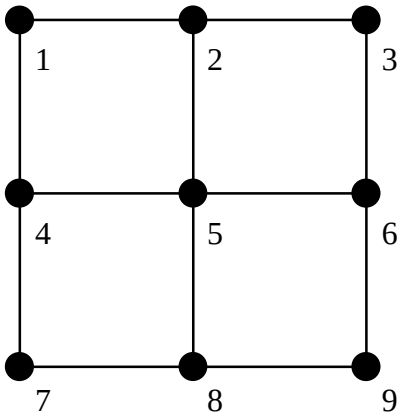
ncont: Number of values in the ncon array (see below)

sehtemp: Flag that is equal to 1 for particle tracking

The following arrays are then read from .stor:

sx1(i), i=1,neq: Primary (total) volume of each node in a continuum (includes fracture and matrix)

ncon(i), i=1,ncont: Node connectivity array that contains the node numbers for each connection to a specified node in one continuum, starting with node 1. The node numbers in ncon associated with connections to a given node include the node of interest. All nodes connected to a given node are listed in ascending order. In the beginning of this array is pointer information with neq+1 entries. The entries identify the index of the array (i=1,ncont) that precedes the node denoted by the index of the pointer information. See the figure below for an example of a 9-node network.



index i	ncon(i)	
1	10	Pointer Information
2	13	
3	17	
4	20	
5	24	
6	29	
7	33	
8	36	
9	40	
10	43	
11	1	Node 1
12	2	
13	4	
14	1	Node 2
15	2	
16	3	
17	5	
18	2	Node 3
19	3	
20	6	
21	1	Node 4
22	4	
23	5	
24	7	
25	2	Node 5
26	4	
27	5	
28	6	
29	8	
30	3	Node 6
31	5	
32	6	
33	9	
34	4	Node 7
35	7	
36	8	
37	5	Node 8
38	7	
39	8	
40	9	
41	6	Node 9
42	8	
43	9	

9-Node Example of the ncon Array Used in FEHM.

istrw(i), i=1,ncont: Not used in this application. The array is filled using following algorithm:

```
do i = 1, ncont
      if(i.le.iwtotl) then
            istrw(i) = i
      else
            istrw(i) = 0
      end if
end do
```

nelmdg(i), i=1,neq: Position (index) of node i in the ncon array:

```
do i = 1, neq
      do j = ncon(i) + 1, ncon(i+1)
            if (ncon(j).eq.i) nelmdg(i) = j
      end do
end do
```

iwtotl numbers: Three groups of iwtotl numbers signifying the x, y, and z components of the nodes are divided by distance terms for all internode connections. Only place-holders are required:

```
do i = 1,3
      write(15,'(5(1pe16.8))') (-1.0, j=1, iwtotl)
end do
```

The file 'fmsd9_e9.stor' is shown below:

```
"fmsd9_e9.stor" 98 lines, 6309 characters
 # input file for mean alpha, fitted fmx, present day q, ysw # AR 11/19/97
This is a .stor file with dummy area coefficients
      73        25        99         1
  4.93991416E+04  4.94314381E+04  3.02836879E+04  6.70661157E+04  8.54244306E+04
  3.21230769E+04  1.43904899E+05  1.97772021E+05  4.78251121E+04  2.86899225E+05
  5.73809524E+05  2.86935484E+05  1.67386018E+05  1.91306991E+05  2.86899696E+05
  2.15197568E+05  2.15187970E+05  2.67276423E+03  1.84173669E+04  1.26272727E+05
  1.48363636E+05  1.48363636E+05  1.45614035E+05  1.50909091E+05  1.90909091E+05
      26        28        31        34        37
      40        43        46        49        52
      55        58        61        64        67
      70        73        76        79        82
      85        88        91        94        97
      99         1         2         1         2
       3         2         3         4         3
       4         5         4         5         6
       5         6         7         6         7
       8         7         8         9         8
       9        10         9        10        11
      10        11        12        11        12
      13        12        13        14        13
```

```
      14        15        14        15        16
      15        16        17        16        17
      18        17        18        19        18
      19        20        19        20        21
      20        21        22        21        22
      23        22        23        24        23
      24        25        24        25
       1         2         3         4         5
       6         7         8         9        10
      11        12        13        14        15
      16        17        18        19        20
      21        22        23        24        25
      26        27        28        29        30
      31        32        33        34        35
      36        37        38        39        40
      41        42        43        44        45
      46        47        48        49        50
      51        52        53        54        55
      56        57        58        59        60
      61        62        63        64        65
      66        67        68        69        70
      71        72        73         0         0
       0         0         0         0         0
       0         0         0         0         0
       0         0         0         0         0
       0         0         0         0         0
       0         0         0         0
      27        30        33        36        39
      42        45        48        51        54
      57        60        63        66        69
      72        75        78        81        84
      87        90        93        96        99
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00
-1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
```

```
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
 -1.00000000E+00 -1.00000000E+00 -1.00000000E+00
```

Output File '*.zone'

This file contains definitions of zones that correspond to ROCKS materials in TOUGH2. The materials are listed sequentially in the same order as they appear in the ROCKS card. A comment (#) is added to identify the name of the material as it appears in ROCKS. The number of nodes within each zone is listed after the header 'nnum'. Following that line, the nodes are listed in the order that they appear in the ELEME card in TOUGH2. Note that both the fracture and matrix materials are listed in this file. Additional comments are added after the 'stop' line of the file. The first few lines of 'fmsd9_e9.zone' are shown below:

```
zone
   1     #tcwM1
nnum
         1
        26
   2     #tcwM2
nnum
         1
        27
   3     #tcwM3
nnum
         1
        28
```

Output File '*.zone2'

This file is identical to the .zone file except that it contains two additional zones that define the repository nodes for the fractures and matrix. The repository elements are listed in another file that is specified by the user during one of the prompts by T2FEHM2. This external file should contain the total number of repository elements in the file followed by a line-by-line listing of all the repository elements. This zone (.zone2) is read at the end of the .dat file to identify nodes where particles will be released in the ptrk macro (note that ptrk is not created by this post-processor). The nodes that are defined in .zone2 will retain the porosities and densities assigned to them previously in .rock and '.dpdp.' For the 1-D FEHM simulations in this analysis, only one repository element ('FlE71') is identified. The last few lines from 'fmsd9_e9.zone2' are shown below:

```
500   #fracture repository nodes
nnum
         1
        12
 501   #matrix repository nodes
nnum
         1
        37

stop


#Total number of nodes =       50
#Total number of active boundary materials =        2
#Total number of active boundary nodes =        2
```

Verification:

The output from T2FEHM2 has been verified by visual inspection in the previous section that detailed the output files. This ensures that T2FEHM2 is performing correctly for the range of inputs used in this analysis. All files relevant to T2FEHM2 can be found in DTN: SN9908T0581699.001.

Listing of Software Routine T2FEHM2 v. 2

```
c     t2fehm2_v2.f
C*****************************************************************
c     This program creates column formatted files from TOUGH2.OUT
c     files of EOS3 simulations.
c     Files  MESH, TOUGH2.INP, and TOUGH2.OUT must be present.
c     The format of the output files are amenable for an FEHM
c     restart.
c                         C.K.Ho 5/27/97
c  This version now re-formats TOUGH2.OUT files in either EOS3 or
c  EOS9 format. Multidimensional files can be post-processed. This
c  version assumes that the elements listed in ELEME alternate
c  between fractures and matrix, starting with a fracture element.
c  This can be generalized in the loop (do 3000...) by knowing how
c  how the fracture and matrix elements were listed and by arranging
c  the arrays accordingly.  I started this by asking the user to
c  specify the ordering, but I didn't do much with it in this version.
c  So for now, the elements should be listed alternately starting with
c  a fracture element. Also, the matrix materials are assumed to be listed
c  first in the ROCKS card.
c
c     C.K.Ho
c     9/2/97-9/12/97,9/19/97
c  This version (op1postv3.f) is tailored specifically for LBL site-scale
c  runs. The previous version (option1postv2.f) is still good for SNL
c  TOUGH2 simulations of flow fields.  The major revisions include reading
c  information from external files (MESH, GENER). In MESH, the material
c  identifier is a 5-character name--not an integer, which was assumed in
c  the previous version.  The coordinates will have to be
```

```
c  read from MESH. Changes will have to be made for recognizing
c  fracture or matrix materials to accomodate all the materials (there
c  are greater than 100 materials) in the site-scale model. The dimensions
c  will have to be greatly increased to accommodate the 80,000 element
c  site-scale model.
c     C.K.Ho
c     10/23/97
c
c  This version (op1postv4.f) does not assume any ordering in the ROCKS
c  card.  There can be different numbers of matrix and fracture
c  materials written to the FEHM zone macro.  Also, this version can read
c  in a file containing repository element names to create a separate zone.
c  Another assumption is that the active elements are listed before any
c  boundary elements ('TP' or 'BT') in ELEME.
c     C.K.Ho
c     11/5/97
c
c  A few things have been cleaned up and it appears to work for the LBNL
c  3-D site scale model.  The current version is 't2fehm2.f'.
c     C.K.Ho
c     11/6/97
c
c  This version accommodates new output formatting used by LBNL.  The
c  index field in the output has been changed from i6 to i12.  Also,
c  the flux output has been shifted to the left a bit, and nlin3 is now equal
c  to 3 instead of 4 (this is the amount of header lines inserted in the flux
c  output periodically).
c  The liquid pressure now appears where the gas pressure used to appear in
c  the output file.  To calculate the gas pressure: Pg=Pl-Pc
c     C.K.Ho
c     3/9/99
c*****************************************************************
c234567890123456789012345678901234567890123456789012345678901234567890123456789012
C
      implicit double precision (a-h,o-z)
      DIMENSION X(99000),Y(99000),z(99000),SL(99000),vol(99000)
      dimension PG(99000)
      dimension gelem(99000),ifm(99000)
      dimension fluxl(990000),fmlfm(99000),ncord(99000)
      dimension icon2(990000),flol2(990000),istrw(990000)
      dimension drok(500),por(500),nelmdg(99000),ncon2(99000)
      double precision lblpor
      CHARACTER*22 BLOCK
      CHARACTER*5  ELEMN(99000),ELEM1(490000),ELEM2(490000),ELEMX
      character*5  genname,matname(500),matb,mat(99000)
      character*80 header
      character*40 filen,control,dat,grid,ini,stor,dpdp,rock,zone
      character*40 filein,fileout,meshfile,repfile,zone2,check
      character*1 char2
      character*5 repname(1003)
      common/int/ ncon(99000),icon(99000,35)
      common/flux/ flol(99000,35)
C
      write(*,*) 'This program will re-format TOUGH2 output files'
      write(*,*)'for FEHM restart files. The following files'
      write(*,*)'must be present: input, output, and MESH.'
      write(*,*)'The MESH file should contain 5-character material'
      write(*,*)'names.'
      write(*,*)
      write(*,*)'What is the name of the input file?'
      read(*,*) filein
      write(*,*)'What is the name of the output file?'
      read(*,*) fileout
```

```
      write(*,*)'What is the name of the MESH file?'
      read(*,*) meshfile
      write(*,4)
4     format('What type of run is this?'/,'1) SNL EOS3'/,'2) SNL EOS9?'/
     & ,'3) LBNL EOS9'/,'4) LBNL EOS9 SR/LA')
      read(*,*) neos
      write(*,*)'What reference name would you like to use for the'
      write(*,*)'FEHM restart files? (no spaces in the name)'
      read(*,*) filen
      write(*,*)'In ELEME, how are the elements listed?'
      write(*,*)'(1) Alternatively with matrix first'
      write(*,*)'(2) Alternatively with fracture first'
      write(*,*)'(3) All matrix, then all fractures'
      write(*,*)'(4) All fractures, then all matrix'
      read(*,*) norder
      write(*,*)'For fracture-matrix connections, which element is'
      write(*,*)'listed first:  (1) Fracture or (2) Matrix?'
      read(*,*) nfmc
      write(*,*)'What is the print-out time (sec) of interest?'
      read(*,*) tsec
      write(*,*)'The fracture volumes will be used as the primary'
      write(*,*)'control volume for each element.  Have they been'
      write(*,*)'modified in TOUGH2.INP? (1=yes,  0=no)'
      read(*,*) nvol
      volscale=1.
      if(nvol.eq.1) then
        write(*,*)'What is the scaling factor to retrieve correct',
     &  ' primary volumes from fracture volumes?'
        read(*,*) volscale
      end if
      write(*,7)
7     format('What is the geometry?'/'0) 3-D'/'1) X-Y Plane'/
     &       '2) X-Z Plane'/'3) Y-Z Plane')
      read(*,*) icnl
      write(*,*)'Is there a file with repository element names?'
      write(*,*)'1 = yes,  0 = no'
      read(*,*) nrepans
      if(nrepans.eq.1) then
        write(*,9)
9     format('What is the name of the file with repository elements?')
        read(*,*) repfile
        write(*,*)'Would you like to modify the 2nd character of the'
        write(*,*)'element name?  1=yes,  0=no'
        read(*,*) n2nd
        if(n2nd.eq.1) then
          write(*,*)'What character would you like to use?'
          read(*,'(a1)') char2
        end if
        open(19,file=repfile,status='old')
      end if

      if(norder.eq.1.or.norder.eq.2) then
        nalt=2
      else
        nalt=1
      end if

c...Define FEHM restart files based on reference name
      kend=index(filen,' ')
      control=filen(1:kend-1)//'.files'
      dat=filen(1:kend-1)//'.dat'
      grid=filen(1:kend-1)//'.grid'
      ini=filen(1:kend-1)//'.ini'
```

```
      stor=filen(1:kend-1)//'.stor'
      dpdp=filen(1:kend-1)//'.dpdp'
      rock=filen(1:kend-1)//'.rock'
      zone=filen(1:kend-1)//'.zone'
      zone2=filen(1:kend-1)//'.zone2'
      check=filen(1:kend-1)//'.check'

      if(neos.eq.1) then
        nlin1=5
        nlin2=3
        nlin3=3
      elseif(neos.eq.2) then
        nlin1=6
        nlin2=4
        nlin3=4
      elseif(neos.eq.3) then
        nlin1=6
        nlin2=3
        nlin3=4
      elseif(neos.eq.4) then
        nlin1=6
        nlin2=3
        nlin3=3
      end if

      write(*,*) 'Thank You!  Please wait while I work...'
      open(1,file=meshfile,status='old')
      open(2,file=fileout,status='old')
      open(3,file=filein,status='old')
      open(11,file=control,status='unknown')
      open(12,file=dat,status='unknown')
      open(13,file=grid,status='unknown')
      open(14,file=ini,status='unknown')
      open(15,file=stor,status='unknown')
      open(16,file=dpdp,status='unknown')
      open(17,file=rock,status='unknown')
      open(18,file=zone,status='unknown')
      open(22,file=check,status='unknown')
      open(23,file=zone2,status='unknown')

c....Data
      spht=1.e3
      per1=1.e-15
      per2=1.e-15
      per3=1.e-15
      day=365.25e6
      tims=365.25e6
      nstep=10
      iprtout=10
      iyear=1997
      month=10
      maxit=-10
      epm=1.e-4
      north=40
      ja=1
      jb=0
      jc=0
      igaus=1
      as=1.
      grav=3.
      upwgt=1.
      iamm=5
      aiaa=2.
```

```
      daymin=1.e-10
      daymax=1.e10
      lda=1
      g1=1.e-5
      g2=1.e-5
      g3=1.e-5
      tmch=-1.e-4
      overf=1.2
      irdof=0
      islord=0
      iback=0
      icoupl=0
      rnmax=14400.
      ntt=1
      intg=-1
      zero=1.d-10
      ra=287.
      rv=461.52
C
c...Read header from TOUGH2.INP
      read(3,'(a80)') header

c_____________________________________________________________________
c...Write information to .dat file
      write(12,510) header
510   format(a80/'# Particle tracking for TOUGH2 flow field')

c...Write dpdp macro
      write(12,516) dpdp
516   format('dpdp'/'file'/a)

c...Write perm macro
      write(12,518) per1,per2,per3
518   format('perm'/'1  0  0  ',3e10.3/)

c...Write rlp macro
      write(12,520)
520   format('rlp'/'1  0.  0.  1.  1.  0.  1.'//'1  0  0  1'/)

c...Write rock macro
      write(12,522) rock
522   format('rock'/'file'/a)

c...Write flow macro
      write(12,524)
524   format('flow'/)

c...Write time macro
      write(12,526) day,tims,nstep,iprtout,iyear,month
526   format('time'/2e13.5,4i8/)

c...Write ctrl macro
      write(12,528) maxit,epm,north,ja,jb,jc,igaus,as,grav,upwgt,
     & iamm,aiaa,daymin,daymax,icnl,lda
528   format('ctrl'/i8,e10.2,i8/4i8/'0'/3f10.2/i8,3e10.2/2i8)

c...Write iter macro
      write(12,530) g1,g2,g3,tmch,overf,irdof,islord,iback,icoupl,
     & rnmax
530   format('iter'/5e10.2/4i8,e10.2)

c...Write sol macro
      write(12,532) ntt,intg
```

```
532   format('sol'/2i8)

c...Write rflo macro
      write(12,534)
534   format('rflo'/'air'/'-1'/'20.0  0.1')

c...Write node macro
      write(12,536)
536   format('node'/'1'/'1')

c...Write zone macro that corresponds to the repository nodes
      write(12,515) zone2
515   format('zone'/'file'/a)

c...Write ptrk macro
      write(12,538) filen(1:kend-1)
538   format('ptrk'/'file'/a,'.ptrk')

c...Write stop
      write(12,540)
540   format('stop')
c_____________________________________________________________

c...Write information to control file
      write(11,501) dat,grid,zone,filen(1:kend-1),ini,filen(1:kend-1)
     &,filen(1:kend-1),filen(1:kend-1),filen(1:kend-1),stor,
     &filen(1:kend-1)
501   format(a/a/a/a,'.out'/a/a,'.fin'/a,'.his'/a,'.trc'/a,'.con'//
     & a/a,'.chk'/'all'/'0')

c...Read in repository element names
      if(nrepans.eq.1) then
        read(19,*) nrepelem
        numrep=nrepelem
        do i=1,nrepelem
          read(19,'(a5)') repname(i)
          repname(i)(1:1)='F'
          if(n2nd.eq.1) repname(i)(2:2)=char2
        end do
      end if

c...Read in grid information from MESH
      nbelm=0
      nbmat=0
      matb='     '
      N=1
      read(1,1000) block
1000  format(a22)
99    read(1,65) elemn(n),mat(n),vol(n),x(n),y(n),z(n)
65    format(a5,10x,a5,e10.4,20x,3e10.4)
      if(elemn(n).eq.'     ') go to 98
      if(elemn(n)(4:4).eq.'0') elemn(n)(4:4)=' '
c...Count number of boundary elements, nbelm, and number of boundary
c...materials, nbmat.
      if(elemn(n)(1:2).eq.'TP'.or.elemn(n)(1:2).eq.'BT') then
        nbelm=nbelm+1
        if(mat(n).ne.matb) then
          nbmat=nbmat+1
          matb=mat(n)
        end if
      end if
      N=N+1
      GO TO 99
```

```
98    CONTINUE
      NMAX = N - 1
c...NMAX is the total number of elements read from MESH
      write(*,107) nmax
107   format('Have read in ',i8,' elements from MESH...')
c...nnodes is the total number of active nodes
      nnodes=nmax-nbelm

c...Find maximum number of materials used in ROCKS (nmat)
c      nmat=0
c      do i=1,nmax
c        nmat=max(mat(i),nmat)
c      end do
c      write(*,222) nmat
c222   format('Maximum number of active materials = ',i8,'...')

c...nfmat is the number of fracture materials
c      nfmat=(nmat-nbmat)/2

c...Read in connection information from MESH
      N=1
      READ(1,1500) BLOCK
1500  FORMAT(A22,3X,25X,E10.4)
199   read(1,1502) elem1(n),elem2(n),ifm(n)
c...ifm(n) is a flag in the 75th column of the CONNE card that Yu-Shu has
c...specified as equal to '2' for fracture-matrix connections
1502  format(2a5,64x,i1)
      IF(elem1(n)(1:5).EQ.'     '.OR.elem1(n)(1:3).EQ.'+++') GO TO 198
      if(elem1(n)(4:4).eq.'0') elem1(n)(4:4)=' '
      if(elem2(n)(4:4).eq.'0') elem2(n)(4:4)=' '
      N=N+1
      GO TO 199
198   CONTINUE
      NCMAX = N - 1
c...NCMAX is the total number of connections read from MESH
      write(*,203) ncmax
203   format('Have read in ',i8,' connections from MESH...')

c...Read in ROCKS information from TOUGH2 input file
18    read(3,1000) block
      if(block(1:5).ne.'ROCKS') go to 18

      i=1
      nfmat=0
      nmmat=0
408   read(3,410) matname(i),drok(i),por(i)
410   format(a5,5x,2e10.4)
      if(matname(i).eq.'REFCO') go to 408
      if(matname(i).eq.'     ') then
c...ntotmat is the total number of materials in the ROCKS card
c...nmat is the number of materials associated with non-boundary
c...elements
        ntotmat=i-1
        nmat=ntotmat-nbmat
        go to 27
      end if
c...LBNL uses columns 71-80 in the second line of each material card to
c...identify the fracture porosity
      read(3,415) lblpor
415   format(70x,e10.4)
c...nfmat is the total number of fracture materials
      if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F') then
        nfmat=nfmat+1
```

```
        if(neos.eq.3.or.neos.eq.4) por(i)=lblpor
c...The perched water fractures do not have porosities listed in ROCKS.
c...Yu-Shu said that they have the same porosity as the zeolitic fractures,
c...which is 1.1e-5 (phone message 10/31/97).
        if(por(i).eq.0.) por(i)=1.1d-5
      end if
c...nmmat is the total number of matrix materials
      if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M') nmmat=nmmat+1
      read(3,*)
      read(3,*)
      i=i+1
      go to 408

27    continue

c...10/27/97  Ho

c...Write grid macro file
      write(13,202) nnodes/2
202   format('coor'/i8)
c...This assumes that all boundary elements ('TP' and 'BT') are listed
c...after the active elements in ELEME
      do i=1,nnodes/2
        write(13,204) i,x(i*nalt),y(i*nalt),z(i*nalt)
204     format(i8,3(3x,f10.2))
      end do
      write(13,206)
206   format(/'elem'/'2  1'/'1  2  1'//'stop')

c...Initialize generation array
      do i=1,nmax
        gelem(i)=0.
      end do

c...Read in generation information from TOUGH2.INP
      i=1
33    read(3,1000,end=299) block
      if(block(1:5).ne.'GENER') go to 33
74    read(3,75) genname,g
75    format(a5,35x,e10.4)
      if(genname.eq.'     ') go to 77
      if(genname(4:4).eq.'0') genname(4:4)=' '
      do ik=1,nmax
       if(genname.eq.elemn(ik)) then
c...Assign a generation term for each element (flow into an element
c...is defined as negative)
c...The method used here is different than in v3.  It eliminates a
c...separate do-loop and the need for arrays igen and g.
         gelem(ik)=-g
         i=i+1
         go to 74
       end if
      end do
      write(*,*)'Could not find element name for generation'
      write(*,79) i,genname
79    format('element ',i8,': ',a5)
      stop

299   write(*,*)'***Warning*** No generation card in TOUGH2.INP'

77    ngentot=i-1

c...Write zone macro
```

```
      ntotin=0
      write(18,'(a4)') 'zone'
      write(23,'(a4)') 'zone'
      do i=1,ntotmat
        write(18,512) i,matname(i)
        write(23,512) i,matname(i)
512     format(i4,5x,'#',a5)
        write(18,'(a4)') 'nnum'
        write(23,'(a4)') 'nnum'
        nin=1
        do j=1,nmax
c...Match nodes to respective materials. This assumes that the
c...fractures and matrix elements are listed alternately in ELEME
c...starting with the fractures first
c...If element is a boundary element, go to next element
          if(elemn(j)(1:2).eq.'TP'.or.elemn(j)(1:2).eq.'BT') goto 517
          if(mat(j).eq.matname(i)) then
            if(mat(j)(3:3).eq.'F'.or.mat(j)(4:4).eq.'F') then
               ncord(nin)=(j+1)/nalt
               nin=nin+1
               go to 517
            end if
            if(mat(j)(3:3).eq.'M'.or.mat(j)(4:4).eq.'M') then
              ncord(nin)=j/nalt+nnodes/2.
              nin=nin+1
            end if
          end if
517     end do
        nin=nin-1
        ntotin=ntotin+nin
        write(18,'(i10)') nin
        write(23,'(i10)') nin
        if(nin.gt.0) write(18,'(8i10)') (ncord(k),k=1,nin)
        if(nin.gt.0) write(23,'(8i10)') (ncord(k),k=1,nin)
      end do
      write(18,*)
      write(18,'(a4)') 'stop'

c...Now write zones for nodes corresponding to repository elements
      nrp=1
      do i=1,nmax
        do j=1,numrep
          if(elemn(i).eq.repname(j)) then
            ncord(nrp)=(i+1)/nalt
            nrp=nrp+1
            go to 527
          end if
        end do
527   end do

      nrp=nrp-1
      write(23,*) '500   #fracture repository nodes'
      write(23,'(a4)') 'nnum'
      write(23,'(i10)') nrp
      if(nrp.gt.0) write(23,'(8i10)') (ncord(k),k=1,nrp)
      write(23,*) '501   #matrix repository nodes'
      write(23,'(a4)') 'nnum'
      write(23,'(i10)') nrp
      do i=1,nrp
        ncord(i)=ncord(i)+nnodes/2.
      end do
      if(nrp.gt.0) write(23,'(8i10)') (ncord(k),k=1,nrp)
      write(23,*)
```

```
      write(23,'(a4)') 'stop'

c...Now write some additional information to the zone file
      write(18,*)
      write(23,*)
      write(18,514) ntotin,nbmat,nbelm
      write(23,514) ntotin,nbmat,nbelm
514   format(/'#Total number of nodes = ',i8/'#Total number of',
     & ' active boundary materials = ',i8/'#Total number of active',
     & ' boundary nodes = ',i8/)

c...Write dpdp macro file
      write(16,550)
550   format('dpdp'/'1')
c...Loop over the materials and print out fracture porosities
      do i=1,ntotmat
        if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F') then
          write(16,552) -i,jb,jc,por(i)
552       format(3i8,5x,e10.4)
        end if
      end do
      write(16,554) ja,jb,jc
554   format(/,3i8,5x,'99.'//'stop')

c...Write rock macro file
      write(17,556)
556   format('rock')
      do i=1,ntotmat
        porock=por(i)
        if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F')porock=1.
        write(17,558) -i,jb,jc,drok(i),spht,porock
558     format(3i8,5x,e10.4,5x,e10.4,5x,e10.4)
      end do
      write(17,559)
559   format(/'stop')

c...Search for "TOTAL TIME" in TOUGH2.OUT and then read in variables
89    READ(2,1000,END=90) BLOCK
      IF(BLOCK(1:12).NE.'  TOTAL TIME') GO TO 89
      READ(2,1001) TIME
      if(time.ne.tsec.and.tsec.gt.0) go to 89
1001  FORMAT(E13.5)
      do nl=1,nlin1
        READ(2,1000) BLOCK
      end do
C
c234567890123456789012345678901234567890123456789012345678901234567890123456789012

c...Read in state variables from TOUGH2.OUT
115   N1=1
      N2=MIN(NMAX,45)
      DO 2000 I=N1,N2
      if(neos.eq.1) then
c...    This is EOS3 format
        READ(2,1002) PG(I),SL(I)
1002    FORMAT(12x,e12.5,24x,7e12.5)
      elseif(neos.eq.2.or.neos.eq.3) then
c...    This is EOS9 format
        read(2,118) pg(i),sl(i)
118     format(12x,2e12.5)
      elseif (neos.eq.4) then
c...    This is EOS9 format with new index formatting of i12
        read(2,119) pl,sl(i),pc
```

```
        pg(i)=pl-pc
119     format(18x,3e12.5)
      end if
2000  CONTINUE
C
2100  CONTINUE
c...Check to see if we've read in all the element variables
      IF(N2.EQ.NMAX) GO TO 91
      N1=N2+1
      N2=MIN(NMAX,N1+56)
      do nl=1,nlin2
        READ(2,1000) BLOCK
      end do
      DO 2010 I=N1,N2
      if(neos.eq.1) then
c...    This is EOS3 format
        READ(2,1002) PG(I),SL(I)
      elseif(neos.eq.2.or.neos.eq.3) then
c...    This is EOS9 format
        read(2,118) pg(i),sl(i)
      elseif (neos.eq.4) then
c...    This is EOS9 format with new index formatting of i12
        read(2,119) pl,sl(i),pc
        pg(i)=pl-pc
      end if
2010  CONTINUE
      GO TO 2100
C
91    CONTINUE
C
c...Write saturations to .ini file (fractures saturations first followed
c...by matrix saturations)
      write(14,302) header
302   format(a80/'This is a .ini file with saturations, pressures',
     & ' and mass flux values.'/'0.'/'air'/'ptrk'/'nstr'/
     & 'dpdp'/'ndua')
      write(14,304) (sl(i),i=1,nnodes,2),(sl(i),i=2,nnodes,2)
304   format(4g16.8)

c...Write pressures to .ini file in MPa (fractures first, then matrix)
      write(14,304) (pg(i)*1.d-6,i=1,nnodes,2),
     & (pg(i)*1.d-6,i=2,nnodes,2)

      write(*,*)'Have read in state variables from output file...'
C
c...Read in flux variables from TOUGH2.OUT
289   READ(2,1500,END=190) BLOCK
      if(neos.lt.4) then
        IF(BLOCK(11:22).NE.'ELEM1  ELEM2') GO TO 289
      elseif (neos.eq.4) then
        IF(BLOCK(7:18).NE.'ELEM1  ELEM2') GO TO 289
      end if
      READ(2,1500) BLOCK
      READ(2,1500) BLOCK
C
c...Read in mass flow liquid for each connection pair
      N1=1
      N2=MIN(NCMAX,53)
      DO 1600 I=N1,N2
      if(neos.eq.1) then
        READ(2,1003) fluxl(I)
1003    FORMAT(80x,4e13.5)
      elseif (neos.eq.2.or.neos.eq.3) then
```

```
        read(2,121) fluxl(i)
121     format(29x,e13.5)
      elseif (neos.eq.4) then
        read(2,122) fluxl(i)
122     format(31x,e13.5)
      end if
1600  CONTINUE
C
2150  CONTINUE
      IF(N2.EQ.NCMAX) GO TO 191
      N1=N2+1
      N2=MIN(NCMAX,N1+56)
      do nl=1,nlin3
        READ(2,1500) BLOCK
      end do
      DO 2020 I=N1,N2
      if(neos.eq.1) then
        READ(2,1003) fluxl(I)
      elseif (neos.eq.2.or.neos.eq.3) then
        read(2,121) fluxl(i)
      elseif (neos.eq.4) then
        read(2,122) fluxl(i)
      end if
2020  CONTINUE
      GO TO 2150
C
191   CONTINUE

C
190   CONTINUE

c...Check
      write(*,*)'Have read in flux variables from output file...'

c...Check
c      do i=1,ncmax
c        write(15,444) i,elem1(i),elem2(i),fluxl(i)
c444     format(i8,2x,2(a5,2x),e10.4)
c      end do
c      stop
c...End check
C
c...Loop over all elements to determine connections and fluxes for each
c...element
      nmlfm=1
c...nmlfm is the total number of fracture-matrix connections
      DO 3000 I=1,NMAX

       if(mod(i,1000).eq.0) write(*,472) i
472    format('Still working...  Element ',i8)

c...fmlfm(i) is the flow (kg/s) between fracture and matrix
       fmlfm(i)=0.d0

c...jj is the number of connections for each element
       do jj=1,35
         flol(i,jj)=0.d0
c...icon(i,jj) is the node number of the element for connection jj to element i
         icon(i,jj)=0
       end do

      ELEMX=ELEMN(I)
```

```
c...If element is a boundary element, go to next element
      if(elemx(1:2).eq.'TP'.or.elemx(1:2).eq.'BT') go to 3000

c...Write the element number and the number of connections for that element
      if(i.gt.1) write(22,*) i-1,ncon(i-1)
c_____________________________________________________________________
c...For each element, loop over all connections to determine if
c...the element is either the first or second element in each connection
c...nc is the number of connections per element

      nc=1
      DO 3001 J=1,NCMAX

c...Say element is the first element in the connection
       if(elem1(j).eq.elemx) then
        nsign=-1
c...If connecting element is the top boundary, go to next connection
        if(elem2(j)(1:2).eq.'TP') go to 3001
c...If connecting element is the bottom boundary, treat the flow to the
c...bottom boundary as a sink/source term and move on to the next connection
        if(elem2(j)(1:2).eq.'BT') then
          gelem(i)=fluxl(j)*nsign
          go to 3001
        end if
c...What is the second element in the connection?
        do ii=1,nmax
         if(elem2(j).eq.elemn(ii)) then
           k2nd=ii
c...Determine if the connection is between a fracture and matrix element
c...If it is a fracture-matrix connection (both elements have the same
c...coordinates, or ifm=2), store this flux separately from fracture-fracture
c...or matrix-matrix fluxes.
           dx=dabs(x(k2nd)-x(i))
           dy=dabs(y(k2nd)-y(i))
           dz=dabs(z(k2nd)-z(i))
           if(dx.le.zero.and.dy.le.zero.and.dz.le.zero.or.
     &        ifm(j).eq.2) then
c...If the first element of f-m connection is a fracture, then process this
             if(nfmc.eq.1) then
               go to 3017
             else
               go to 3001
             end if
           end if
           icon(i,nc)=ii
           flol(i,nc)=fluxl(j)*nsign
           nc=nc+1
           go to 3002
         endif
        end do
        write(*,7001) elemx,j,elem2(j),elem2(j-1),elem2(j+1)
7001    format('***Could not find 2nd element in connection for',
     &  ' first element ',a5,'***'/'Connection index = ',i8/
     &  'Second element = ',a5/'j-1= ',a5/'j+1= ',a5)
        stop
       end if

c...If no match in first element of connection, try second element
       if(elem2(j).eq.elemx) then
        nsign=1
c...If connecting element is the top boundary, go to next connection
        if(elem1(j)(1:2).eq.'TP') go to 3001
c...If connecting element is the bottom boundary, treat the flow to the
```

```
c...bottom boundary as a sink/source term and move on to the next connection
        if(elem1(j)(1:2).eq.'BT') then
          gelem(i)=fluxl(j)*nsign
          go to 3001
        end if
c...What is the first element in the connection?
        do ii=1,nmax
         if(elem1(j).eq.elemn(ii)) then
          k2nd=ii
c...Determine if the connection is between a fracture and matrix element
c...If it is a fracture-matrix connection (both elements have the same
c...coordinates), store this flux separately from fracture-fracture or
c...matrix-matrix fluxes.
           dx=dabs(x(k2nd)-x(i))
           dy=dabs(y(k2nd)-y(i))
           dz=dabs(z(k2nd)-z(i))
           if(dx.le.zero.and.dy.le.zero.and.dz.le.zero.or.
     &        ifm(j).eq.2) then
c...If the second element of f-m connection is a fracture, then process this
             if(nfmc.eq.2) then
               go to 3017
             else
               go to 3001
             end if
           end if
           icon(i,nc)=ii
           flol(i,nc)=fluxl(j)*nsign
           nc=nc+1
           go to 3002
         end if
        end do
        write(*,7000) elemx,j,elem1(j)
7000    format('***Could not find 1st element in connection for',
     &  ' second element ',a5,'***'/'Connection index = ',i8/
     &  '1st element = ',a5)
        stop
       end if

c...If neither element 1 or 2 for connection j is equal to elemx, then
c...go on to the next connection
       goto 3001

3002   continue

c_____________________________________________________________________
c...go to next connection
       go to 3001
c_____________________________________________________________________
c...Come here if this is a fracture-matrix connection AND the element
c...being considered (elemx=elemn(i)) is a fracture
c...Consider outflow to be positive and
c...that the first element in the connection is a fracture
3017  continue
      fmlfm(nmlfm)=nsign*fluxl(j)
      nmlfm=nmlfm+1

c...Go to next connection
c_____________________________________________________________________
3001  continue

c...ncon(i) is the total number of connections for node i
      ncon(i)=nc-1
C
```

```
c...Check
c        write(15,446) i,ncon(i),(icon(i,j),j=1,ncon(i))
c446     format(10(i8,2x))
c        write(15,448) i,ncon(i),(flol(i,j),j=1,ncon(i))
c448     format(2(i8,2x),8(e10.4,2x))
c...End check

c...Go to next element
3000  CONTINUE

c...nmlfm is the total number of fracture-matrix connections
      nmlfm=nmlfm-1

c...Add connection for each element to itself using generation array
c...nmfluxval is the total number of mass flux values
c...Note: nodes 1-nnodes are still assumed to alternative between
c...fractures and matrix. This will be adjusted later in the print-out
c...to the FEHM files.
      nmfluxval=0
      do i=1,nnodes
        ncon(i)=ncon(i)+1
        icon(i,ncon(i))=i
        flol(i,ncon(i))=gelem(i)
        nmfluxval=nmfluxval+ncon(i)
c...Check
c        write(15,448) i,ncon(i),flol(i,ncon(i)),nmfluxval
c448     format(2(i8,2x),e10.4,2x,i8)
c...End check
c...nmfluxval is the total number of flux values for fracture and matrix
c...elements excluding f-m fluxes
      end do

c...Call sort subroutine to sort the necessary arrays in ascending order
c...of elements for each connection pair of a given element

      call sort(nnodes)
C
c...Create 1-D arrays containing icon and flol information.  The arrays
c...will be icon2 and flol2. This assumes that the fractures and matrix
c...elements alternate in ELEME and fractures are listed first.
      k=1
      jj=1
      ncont1=0
c...ncont1 is the total number of connections for each continuum
c...do the fracture continuum first
      do i=1,nnodes,2
        do j=1,ncon(i)
c...The index k+nnodes/2+1 accounts for the leading pointer information
          icon2(k+nnodes/2+1)=(icon(i,j)+1)/2
          flol2(k)=flol(i,j)
          k=k+1
        end do
        ncont1=ncont1+ncon(i)
c...ncon2(jj) is the number of connections for fracture node jj, where jj is
c...now icremented 1,2,3...nnodes/2
        ncon2(jj)=ncon(i)
        jj=jj+1
      end do

c...Now do the matrix continuum
      do i=2,nnodes,2
        do j=1,ncon(i)
          flol2(k)=flol(i,j)
```

```
          k=k+1
        end do
      end do
c...ntotmfv is the total number of connections.  This can be compared to
c...nmfluxval as a cross-check to see if they're equal.
      ntotmfv=k-1

c...Write mass flux values to .ini file
      write(14,602) nmlfm+nmfluxval,ntotmfv,nnodes,nmlfm
602   format('mass flux values'/i8,5x,'#ntotmfv=',i8,', nnodes=',i8,
     & ', number of f-m connections= ',i8)
      write(14,604) (flol2(i),i=1,ntotmfv),(fmlfm(i),i=1,nmlfm)
604   format(5g15.8)

c...Write .stor file
      write(15,702) header
702   format(a80/'This is a .stor file with dummy area coefficients')

c...Add the pointer information (number of fracture nodes+1) to ncont1
      neq=nnodes/2
      ncont=ncont1+(neq+1)
      iwtotl=ncont-(neq+1)

      write(15,704) iwtotl,neq,ncont,1
704   format(4(i8,2x))

c...Write primary volume for each node to .stor
c...If this is an LBNL run, then divide the fracture volumes by the
c...fracture porosity, since the volumes in ELEME were multiplied by
c...the fracture porosity.
      if(neos.eq.3.or.neos.eq.4) then
        do i=1,nnodes,2
          do j=1,ntotmat
            if(mat(i).eq.matname(j)) then
              vol(i)=vol(i)/por(j)
              go to 833
            end if
          end do
833     end do
      end if
c...If the fracture volumes were globally modified, multiply the volume
c...by a scaling factor, volscale, specified by the user to get the original
c...volume back.
      write(15,706) (vol(i)*volscale,i=1,nnodes,2)
706   format(1p5e16.8)

c...Compile and write ncon and pointer information
c...Fill the icon2(i) array from i=1,neq+1 (recall that icon2(i) has
c...already been filled from neq+2 to ncont1 (the total number of connections
c...for the fracture continuum
      icon2(1)=neq+1
      do i=2,neq+1
        icon2(i)=icon2(i-1)+ncon2(i-1)
      end do
      write(15,708) (icon2(i),i=1,ncont)
708   format(5(i8,2x))

c...Compile and write istrw information to .stor file
      do i=1,ncont
        if(i.le.iwtotl) then
          istrw(i)=i
        else
          istrw(i)=0
```

```
        end if
      end do
      write(15,708) (istrw(i),i=1,ncont)

c...Compile and write nelmdg information to .stor file
      do i=1,neq
        do j=icon2(i)+1,icon2(i+1)
          if(icon2(j).eq.i) nelmdg(i)=j
        end do
      end do
      write(15,708) (nelmdg(i),i=1,neq)

c...Write dummy area coefficients to .stor file
      do i=1,3
        write(15,706) (-1.0,j=1,iwtotl)
      end do
c_____________________________________________________________________

      write(*,1153) time
1153  format('Finished processing printout at ',e12.4,' sec')
      go to 722
C
90    CONTINUE
      write(*,*)'**Did not find desired print-out time in TOUGH2.OUT**'
C
722   write(*,*) 'Done!!!'

      stop
      END


      subroutine sort(nnodes)
c_____________________________________________________________________
c  This subroutine sorts variables using a multipass method.
c     C.K.Ho
c     9/8/97
c_____________________________________________________________________

      implicit double precision (a-h,o-z)
      common/int/ ncon(99000),icon(99000,35)
      common/flux/ flol(99000,35)

c...The objective here is to arrange the connections in ascending order
c...of connecting node number. The associated flux should also be sorted.

      nsort=1
      do i=1,nnodes
5       if(nsort.eq.1) then
          nsort=0
          do j=1,ncon(i)-1
            if(icon(i,j).gt.icon(i,j+1)) then
              itempicon=icon(i,j)
              icon(i,j)=icon(i,j+1)
              icon(i,j+1)=itempicon
              tempflol=flol(i,j)
              flol(i,j)=flol(i,j+1)
              flol(i,j+1)=tempflol
              nsort=1
            end if
          end do
          go to 5
        end if
        nsort=1
```

```
end do
return
end
```

PrepareKDfile V1.0
Routine/Macro Documentation Form

The following information can be included in the scientific notebook. Attach and reference notebook pages and diskettes with files as needed when submitting routine/macro to records.

1. Name of routine/macro with version/OS/hardware environment:
   PrepareKDfile V1.0 (routine) / Windows 98/PC

2. Name of commercial software with version/OS/hardware used to develop routine/macro:
   Digital Visual FORTRAN 5.0 (Fortran 90)/Windows 98/PC

3. **Test Plan.**

   - Explain whether this is a routine or macro and describe what it does:
     This routine is used to read the rock names from the TOUGH2 mesh file and the sorption parameters (rock density and Kd) from the T2R3D input file, and then format the Kd and rock density into the input file for DCPT V1.0.

   - Source code: (including equations or algorithms from software setup (LabView, Excel, etc.):
     pp. 90-91 of YMP-LBNL-GSB-LP-3 (attached)

   - Description of test(s) to be performed (be specific):
     The routine will read the rock density and the Kd parameter for each rock unit from the T2R3D input file "uzm_tr2.dat" (p. 126, YMP-LBNL-GSB-LP-3), and write the two parameters for each cell (fracture and matrix cell are considered as one cell) to the file "UZ99.kd" according the rock types assigned to the cells in the mesh file "MESH_cal.V1". Each row of data in the output file represent a cell which includes a fracture cell and a matrix cell in the mesh file. The cells in the output file are ordered in the same way as those in the mesh file except that both the top and bottom cell for each TOUGH2 column is added. Only cells having nonzero Kd values in the matrix are included. Note that the Kd values for the fracture in the output file will always be zero no matter what values are in the input file.
     By thoroughly comparing the data in the output file "UZ99.kd" with those in the input file "uzm_tr2.dat" the routine will be shown to work properly (YMP-LBNL-GSB-LP-3, pp125-126). To verify the reformatting, a representative sampling of the input and output files will be used. The following method will be used to select this sampling: pick several typical cells in the mesh files; find the corresponding cells in the output file based on their ordering; find the corresponding rock entries in the input file based on the rock names of the picked cells; compare the corresponding rock density and Kd values in the input file and the output file.

   - Specify the range of input values to be used and why the range is valid
     The input data in the original mesh file and the T2R3D input file are used in the related analysis. Therefore, the test case is actually the case that the routine is designed for. Thus, this test case input range is deemed valid. Note that the Kd values for the fractures are always zero. This routine cannot be used if the Kd values for the fractures are not zero.

4. **Test Results.**

   - Output from test (explain difference between input range used and possible input)

The output are several lines from the file "UZ99.kd" printed on pp. 126 of YMP-LBNL-GSB-LP-3. To verify the reformatting, a representative sampling of the input and output files was used.

- Description of how the testing shows that the results are correct for the specified input.
  The reformatting was successful because the comparison of the output and input files on p. 126, YMP_LBNL-GSB-LP-3 shows identical numbers. Also, the test was acceptable because the routine successfully ran without error messages and the new "UZ99.kd" was successfully imported into DCPT V1.0 without errors. Furthermore, the results of DCPT V1.0 are consistent with those of T2R3D V1.4 for the same case. Therefore, the test case and routine are acceptable.

- List limitations or assumptions to this test case and code in general
  The routine was tested using an input data set that is actually used in the related analysis. However, the routine is only valid for the input file of T2R3D V1.4 and the mesh file generated by the WinGridder V1.0. the output file (Kd file) can be used by DCPT V1.0 only. This routine assumes that the Kd values of the fractures are zero. The routine is considered as a single-user routine.

- Electronic files identified by name and location (include disc if necessary)
  The routine, test files and its description can be found on pp. 90-91 and 125-126, from YMP-LBNL-GSB-LP-3.

5. Supporting Information. Include background information, such as revision to a previous routine or macro, or explanation of the steps performed to run the software. Include listings of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

See attached pages for technical review forms, referenced scientific notebook pages and other supporting documentation

Note: All relevant S/N pages are included in this package. In some instances, the included S/N pages cross-reference other pages that are not included here because these were not essential to the documentation of this routine.

MAINTAIN PAGES IN THIS ORDER:

1. **This 2-page routine documentation form.**
2. **pp. 125-126 and 90-91, from YMP-LBNL-GSB-LP-3.**
3. **Review forms**

Principal Investigator

Evaluation on the Initial Entry Compliance Review

The Initial Entry Compliance Review has been completed according to AP-SIII.1Q requirements and all of reviewer's comments have been resolved. An entry has been made in this notebook at the end of the reviewed material indicating completion of this review by reference to the Scientific Notebook Compliance Review Worksheet. The Scientific Notebook Compliance Review Worksheet is a supplemental record to this notebook and it will be submitted to YMP Records Processing Center in accordance with AP-SIII.1Q and AP-17.1Q requirements.

12/22/99

PI Signature Date

LP 12/22/99

12/23/99 Test of "Prepare.KDfile V1.0

For full description of the code, see page 88-90.

The program was run on PC (LBL/DOE 635/820) under Digital Visual Foreran developing environment (MS Developer Studio 97). LP 12/23/99 The input files are: "E:\ParticleTrack\Cal99\uzm_tr2.dat" and "E:\ParticleTrack\Cal99\Mesh_Cal.V1". The output file is "E:\ParticleTrack\V799.kd". LP 12/23/99 LP 12/23/99 The input program is used to extract data from the input file and then save to

SIGNATURE _____ DATE _____ 19 _____

READ AND UNDERSTOOD _____ DATE _____ 19 _____

the output file. No calculation is involved.

— A spot checking is provided below.

(uzm_~~trz.dat~~) LP 12/23/99

The data in the input file (highlighted) are

```
tswM1    2 2510.     0.5300E-010.6321E-160.6321E-160.6321E-16 1.000     1000.
                   0.7200        0.7000
    0.00     0.00        0.0       1.0e-3   1.60e-10
   7    0.3033    0.2200     1.000                1.000
   7    0.3033    0.2200     0.3612E-040.1000E+11 1.000
tswM2    2 2550.     0.1570     0.5825E-150.5825E-150.5825E-15 1.000     1000.
                   0.7200        0.7000
    0.00     0.00        0.0       1.0e-3   1.60e-10
   7    0.3327    0.7000E-01 1.000                1.000
   7    0.3327    0.7000E-010.3807E-040.1000E+11 1.000
tswM3    2 2510.     0.1540     0.3083E-160.3083E-160.3083E-16 1.000     1000.
                   0.7200        0.7000
    0.00     0.00        0.0       1.0e-3   1.60e-10
   7    0.2978    0.1200     1.000                1.000
   7    0.2978    0.1200     0.2125E-040.1000E+11 1.000
```

LP 3/6/00

The corresponding data in the output file (UZ99.Kd) are (highlighted):

```
12 1.000E-03   2510.00 0.000E+00   2510.00
13 1.000E-03   2550.00 0.000E+00   2550.00
14 1.000E-03   2510.00 0.000E+00   2510.00
```

The data are identical numerically.

Test of ExtractFlow V1.0

For full description of the code, see page 91-92. The program ~~is in~~ was running (LP 12/23/99) under Digital Visual Fortran developer environment on ~~a~~ the PC (LBL/DOE 631 7820). (LP 12/23/99)

The input file is "paul-tr1.out" located in "E:\ParticleTrack\Cal99\". The output file

SIGNATURE ______ DATE ______ 19 ___
READ AND UNDERSTOOD ______ DATE ______ 19 ___

PrepareKDfile V1.0

Lehua Pan

Creator
Lehua Pan
3/6/00

```
c This program writes out lehua's kd data from input data and MESH
cc '**
      program xx2
      implicit double precision(a-h,o-z)
      parameter(lmt=60000)
      character rock(lmt)*5
      dimension alfa_l(lmt),alfa_t(lmt),t_half(lmt)
      double precision kd(lmt),diffs(lmt),alfa_fm(lmt),dens(lmt)
      character*5 y1,y3, y2*75, y4*75
      Character*50 RockFile,KDFile,MeshFile,InputDir,OutputDir
      character*3 PreviousColumn
      InputDir="E:\ParticleTrack\Cal99\"
      OutPutDir="E:\ParticleTrack\"
      RockFile=trim(InputDir) // "uzm_tr2.dat"
      meshfile=trim(inputdir) // "MESH_cal.v1"
      KDfile=trim(OUTputDir) // "UZ99.kd"

      kk=0
      i=1
      yy=0.
      yx=0.
      open(11,file=RockFile,status='old', err=30)
!     Lehua's code
      open (16,file=KDfile)
      open (7,file=meshfile,status='old',err=40)
!     end of Lehua's code
cc
14    read(11,'(a5)',end=33) rock(i)
      if(rock(i).ne.'ROCKS'.and.rock(i).ne.'rocks') goto 14
c
4     read (11,'(a5,5x,e10.0)', end=8) rock(i),dens(i)
      if(rock(i).eq.'     '.or.rock(i).eq.'REFCO') goto 8
      read(11,*,end=8)
      read (11,'(7e10.0)', end=8) alfa_l(i),alfa_t(i),
     &    t_half(i),kd(i),diffs(i),alfa_fm(i)
          !Lehua's code
          if (kd(i) .ne. 0. .and. dens(i) .le. 100.) then
              write (*,*) "Small density found! ",rock(i)
              write (*,*) kd(i),dens(i)
              kd(i)=0.
      Write (*,*) "Kd has been set to zero! Press ENTER to continue"
              pause
          endif
      !   End of Lehua's code
      read(11,*,end=8)
      read(11,*,end=8)
      i=i+1
      goto 4
cc
c
8     nn=i-1
      close unit=11
c
!     Lehua's code
      read(7,*) !get rid of the "ELEME"
      kk=kk+1 !account for top cell in mesh used by DCPT
2     continue
      read (7,'(a5,a75)', end=10,err=44) y1,y2 !fracture
      if(y1(1:5).eq.'     ' .or. y1(1:1) .eq. 'T') then !end of file or hit the top cell
          goto 10
      endif
      if (kk .eq. 1) then
          previouscolumn=y1(3:5)
      elseif (y1(3:5) .ne. previousColumn) then !new column
          kk=kk+2  !Account for the bottom cell of the previous column and the top cell of this co
lumn
          previousColumn=y1(3:5)
      endif

      read (7,'(a5,a75)', end=10,err=44) y3,y4 !matrix
      !write (*,*) y1,y2
      !pause
!     end of Lehua's code
      !fracture no adsorption
       kk=kk+1
```


SIGNATURE _____ DATE _____ 19__
READ AND UNDERSTOOD _____ DATE _____ 19__

```
      do m=1,nn
      if(y4(11:15).eq.rock(m) .and. kd(m).ne. 0.) then
      !yy should be kd of fracture but not assigned
        write(16,'(i6,1p,e10.3,0p,f10.2,1p,e10.3,0p,f10.2)')
     &    kk,kd(m),dens(m),yy,dens(m)
        goto 2
      endif
      enddo    !next m
       goto 2
10    continue
      close (16)
      close (7)
      stop "!done"
30    stop "!error"
33    stop "!No ROCKS block!!"
!     Lehua's code
40    stop "!No MESH file!"
44    stop "!error in MESH file!"
!     end of Lehua's code
      end
```

SIGNATURE _____ DATE _____ 19 _____

READ AND UNDERSTOOD _____ DATE _____ 19 _____

YMP-LBNL
REVIEW RECORD

1. QA: L
2. Page 1 of 1

3. Originator: Lehua Pan

4. Document Title: Documentation for Routine **PrepareKDfile v1.0** (Option 1 per AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1)

5. Document Number: N/A 6. Revision/Mod.: N/A 7. Draft: N/A

8. Governing Procedure Number: AP-SI.1Q 9. Revision/Mod: 2/4

REVIEW CRITERIA — YMP-LBNL-QIP-6.1, Attch. 5, p. 18 Routine Review Criteria

10. ☒ Standard Review Criteria (One time use Option 1) (Taken from Attachment 5)

11. ☐ Specific Review Criteria:
☒ Source: AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1.1 (One time use routine)

12. Comment Documentation:
☐ Comment Sheets
☐ Review Copy Mark-up
☐ Attached:
☒ Scientific notebook/data associated with this review as noted on Attachment 3

13. YMP-LBNL Project Manager (PM): Gudmundur S. Bodvarsson

14. Reviewer	Org./Discipline	Review Criteria	Reviewer	Org./Discipline	Review Criteria
R.F. Hedegaard	LBNL/Hydrogeologist	Technical			

COMMENTS DUE:

15. Due Date: 2/18/00

16. Originator/Review Coordinator: Lehua Pan
Print Name

REVIEW BY:

17. Randall F. Hedegaard
Print Name

18. [signature] RFH 3-3-00 3-7-00
Signature Date

19. Mandatory Comments: ☐ Yes ☒ No

CONCURRENCE:

21. Document Draft No: NA Date: -

22. Reviewer: [signature] 3-7-00
Signature Date

23. PM: [signature] 3/7/00
Signature Date

ORIGINATOR/REVIEW COORDINATOR (After response completed):

20. Lehua Pan [signature] 3/6/00
Print Name/Signature Date

DISPUTE RESOLUTION: (if applicable)

24. PM:
Signature Date

YMP-LBNL
COMMENT SHEET

QA: L

1. Document Title: **Routine Documentation for PrepareKDfile v1.0**			2. Page 1 of 1
3. Document No. **N/A**	4. Revision/ Change/Mod: **N/A**	5. Draft **N/A**	6. ☒ Q ☐ NQ
7. Reviewer: **Randall F. Hedegaard**			

8. NO. CODE	9. SECT./PARA./P#	10. COMMENT	11. RESPONSE	12. ACCEPT
		--NO COMMENTS— The documentation for this routine was reviewed and it was found to meet the requirements of AP-SI.1Q/Rev. 2/ICN4. The test case was checked by both hand calculation and by running the code as needed to fully check the test case. The test case fully checks the routine for the input specified and proves that the routine produces acceptable results.		

YMP-LBNL
APPLICABLE REFERENCE INFORMATION

Document No.and Title: Routine Documentation for **PrepareKDfile v1.0** per Option 1, AP-SI.1Q/Rev.2/ICN4, Sec. 5.5.1

Date of Document (or revision, draft revision number, as applicable):
NA

Pertinent sections of scientific notebook(s) or other backup documents and/or data DTN# are identified below, supporting the document which is the subject of this review. These documents/data shall be included in the scope of this review.

Document(s) Title/Data	Relevant Sections/Pages
YMP-LBNL-GSB-LP-3 Scientific Notebook	pp. 125-126 and 90-91

STANDARD REVIEW CRITERIA

Routine/Macro Review Criteria, Option 1

NOTE: Where a checklist item does not apply to the software product, check "N/A".

	Yes	No	N/A	
R/M-1	X			**The information given below is to be documented in the technical product, in which the routine/macro is used to support. Does the routine/macro include:** Name of routine/macro with version/Operating System/hardware environment
R/M-2	X			Name of commercial software used to write the routine/macros with version/Operating System/hardware used to develop it
R/M-3	X			Test Plan • Explanation whether this is a routine or macro and a description of what it does • The source code (this section shall include equations or algorithms form software setup (Labview, Excel, etc.) • Description of test(s) to be performed (be specific) • Specified range of input values to be used and why the range is valid
R/M-4	X			Test Results • Output from test (explain difference between input range used and possible input) • Description of how the testing shows that the results are correct for the specified input • List of limitations or assumptions to this test case (s) and code in general • Electronic files identified by name and location (included if necessary to perform the tests)
R/M-5	X			Supporting Information. Include background information, such as revision to a previous routine or macro or explanation of the steps performed to run the software. Include listing of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

StatSpatial V1.0
Routine/Macro Documentation Form

The following information can be included in the scientific notebook. Attach and reference notebook pages and diskettes with files as needed when submitting routine/macro to records.

1. Name of routine/macro with version/OS/hardware environment:
   StatSpatial V1.0 (routine) / Windows 98/PC

2. Name of commercial software with version/OS/hardware used to develop routine/macro:
   Digital Visual FORTRAN 5.0 (Fortran 90)/Windows 98/PC

3. **Test Plan.**

   - Explain whether this is a routine or macro and describe what it does:
     This routine is used to count the number of particles along a user specified line based on the information in the DCPT V1.0 output file and the user-specified resolutions.

   - Source code: (including equations or algorithms from software setup (LabView, Excel, etc.):
     pp. 3-5 of this form

   - Description of test(s) to be performed (be specific):
     The test of the routine will first build a series of adjacent boxes with user-specified dimensions in 3-D space. It will then check the coordinates of all particles in a DCPT V1.4 output file to count the number of particles that are located within each box. Finally, it will write the results to an output file. The first column of the output file will be the coordinate of the box center while the second column will be the number of particles located in the box. Note that in the y-coordinate direction is not subdivided into boxes and that in this direction the entire domain is considered one cell.
     A thorough check of the output file will be done to make sure the routine works properly. The checking will be done as follows: find the particles whose x-coordinate is between 10.15 and 10.35; then check their z-coordinates to determine which box they should go into and then manually count the number of particles for each box. Finally compare these results with the output file.

   - Specify the range of input values to be used and why the range is valid
     The input for the test case is a representative sampling of a DCPT V1.0 output file printed on p. 120 (top of page), YMP-LBNL-GSB-LP-3. This sample input includes particle z-coordinates that are both in and out of the x-coordinate range specified in the test. This will allow the box selection and particle counting functions of the routine to be tested thoroughly. Thus, this test case input range is deemed valid.

4. **Test Results.**

   - Output from test (explain difference between input range used and possible input)
     The output is several lines from the file "TestStatSpatial.out" printed at the bottom of p. 129 of YMP-LBNL-GSB-LP-3. To verify the reformatting, a representative sampling of the input and output files was used.

   - Description of how the testing shows that the results are correct for the specified input.
     The number of particles counted by the routine matches the manual count as explained on pp. 129-130, YMP-LBNL-GSB-LP-3. Because the routine performs simple less than/greater than type functions and reformatting of large data files, a spot-check for consistency between the data within the input file and the output file is

sufficient to justify that the routine works properly. . Therefore, the test case and routine are acceptable.

- List limitations or assumptions to this test case and code in general
  The routine was tested using a portion of the input data set that is actually used in the related analysis. The routine is only valid for the output file of DCPT V1.0. The routine is considered as a single-user routine.

- Electronic files identified by name and location (include disc if necessary)
  The routine, test files and its description can be found on pp. 128-130, from YMP-LBNL-GSB-LP-3.

5. Supporting Information. Include background information, such as revision to a previous routine or macro, or explanation of the steps performed to run the software. Include listings of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

See attached pages for technical review forms, referenced scientific notebook pages and other supporting documentation

Note: All relevant S/N pages are included in this package. In some instances, the included S/N pages cross-reference other pages that are not included here because these were not essential to the documentation of this routine.

MAINTAIN PAGES IN THIS ORDER:

1. **This 2-page routine documentation form.**
2. **pp. 128-130 from YMP-LBNL-GSB-LP-3**
3. **Review forms**

Lehua Pan, the creator
Lehua Pan
3/6/00

```
      program StatSpatial
      use StateCData
      implicit none
      character*100 FName(3),SourceDir,TargetDir
      integer*4 NP,Cor1,Cor2,MBand

      double precision X(3),DX(3)
      SourceDir="e:\particleTrack\"
      TargetDir=SourceDir

!     fname(1)=trim(sourcedir) // "Analy3Dout.txt"
!     fname(2)=trim(Targetdir)// "ana3D2M.out"
!     fname(3)=trim(Sourcedir) // "analy3d.txt"
!     NP=2000000
 !    Test input start
      fname(1)=trim(sourcedir) // "TestStatSpatial.in"
      fname(2)=trim(Targetdir)// "TestStatSpatial.out"
      fname(3)=trim(Sourcedir) // "analy3d.txt"

      NP=20
!     Test input end

      !Location
      X(1)=10.25
      X(2)=10.25
      X(3)=10.0
      !resolution
      DX(1)=0.2
      DX(2)=100.
      DX(3)=0.1
      MBand=200
      cor2=3 !This variable determine which coordinat is interested (1,2,3)=(x,y,z)
      cor1=1  !1=X, 2=Y, 3=Z

      write (*,*) fname(1),fname(2),fname(3)
      !pause
      !call InputGrid(FName(3))
      call spatial(fname(1),NP,X,DX,Cor1,Cor2,MBand)

      !columnID=FindColumn(10.25D0,10.25d0)
      !write (*,*) columnID ," before call"
      call Writespatial(fname(2),MBand)
      !write (*,*) "Column (",columnID,") has been written to"
      write (*,*) trim(fname(2)),", Total NP=",NP,"."
      write (*,*) "Thank you! Bye bye!"

      stop
      end
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      Subroutine WriteSpatial(PRfile,MBand)
      use StateCData
      implicit none
      character*100 PRFile
      integer*4 i,MBand


      open (7,file=prfile)

      do i=1,2*MBand

          write (7,100) PlotArray(i).x,plotArray(i).np



      enddo
      close (7)


      return
100   format (e15.5,I8)
      end
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      Subroutine Spatial(PTfile,NP,T,DT,Cor1,Cor2,MBand)
      !calculate the spatial distribution of the particles
```

~~1~~ 3   LP 3/6/00

```
      !along Cor2
      use StateCData

      implicit none
      character*100 PTFile
      integer*4 i,NP,Cor1,Cor2,MBand, Mmid
      double precision T(3),DT(3),X(3),MaxX
      integer iJunk
      double precision DJunk,HalfDT
      character*2 Cjunk

      type (PlotPoint)::PP

          write (*,*) ptfile
      !initialize
      Mmid=MP/2
      MaxX=T(cor2)+MBand*dt(cor2)


      do i=1,2*MBand
          plotArray(i).x=MaxX-(i-.5)*DT(cor2)

      enddo



      open (7,file=PTfile)
      HalfDT=0.5*DT(cor1)
      !MinX=MinX-0.5*DT(cor2) !adjust half
      do i=1,NP
              read (7,100) Djunk,Cjunk,Djunk,Djunk,
     1 X,ijunk,ijunk

          if (abs(X(cor1)-T(cor1)) .le. halfDT) then
              IJunk=int((MaxX-X(cor2))/DT(cor2)+0.5)
      !write (*,*) X(cor2),minX,DT(cor2),iJunk,cor1,cor2
      !pause

          plotarray(ijunk).np=plotarray(ijunk).np+1
          endif

          !read (7,*) ap

      !write (*,*) ap

          if (mod(i,2000) .eq. 0) then
              write (*,*) i," particles recorded..."
          endif
      enddo
      close (7)
      return
100   format (F10.4,A2,5E15.5,2I8)
      end
```

3/6/01

```
module StateCData
integer*4 MP
parameter (MP=10000)
type PlotPoint
    Double precision X
    integer*4 NP
end type PlotPoint
type (PlotPoint)::PlotArray(MP)
target PlotArray

end module StateCData
```


LP 3/6/00

LP 2/3/00

2/3/2000

Verification of "StatSpatial V1.0"

The detailed descriptions and code lists of "StatSpatial V1.0" are on pages 76-79 of this notebook.

All involved files are in E:\particleTrack\ on PC (LBL/DOE 635720).

Steps:

① Set the input and output files in the code.

LP 2/3/00

```
!     Test input start
      fname(1)=trim(sourcedir) // "TestStatSpatial.in"
      fname(2)=trim(Targetdir)// "TestStatSpatial.out"
      fname(3)=trim(Sourcedir) // "analy3d.txt"

      NP=20
!     Test input end
```

Keep the other parameters same as those used the test case of DCPT:

```
!Location
X(1)=10.25
X(2)=10.25
X(3)=10.0
!resolution
DX(1)=0.2
DX(2)=100.
DX(3)=0.1
MBand=200
cor2=3 !This variable determine which coordinat is interested (1,2,3)=(x,y,z)
cor1=1  !1=X, 2=Y, 3=Z
```

The input file is simply a section of a DCPT output file. It consists of 20 particles. Among them, only 7 particles is located between X = [10.15] and [10.35] (X = 10.25 ± 0.1) which are checked on the print-out. LP

SIGNATURE _____ DATE _____ 19 ____

READ AND UNDERSTOOD _____ DATE _____ 19 ____

			X	Y	Z		
0.0000 I	0.00000E+00	0.10000E+02	0.98804E+01	0.99339E+01	0.95007E+01	1282	1240
0.0000 I	0.00000E+00	0.10000E+02	0.10163E+02 ✓	0.96469E+01	0.11348E+02 *	1282	1299
0.0000 I	0.00000E+00	0.10000E+02	0.10251E+02 ✓	0.10295E+02	0.10190E+02	1282	1302
0.0000 I	0.00000E+00	0.10000E+02	0.10990E+02	0.96191E+01	0.83645E+01	1282	1368
0.0000 I	0.00000E+00	0.10000E+02	0.10165E+02 ✓	0.10288E+02	0.84795E+01	1282	1305
0.0000 I	0.00000E+00	0.10000E+02	0.10720E+02	0.10453E+02	0.92293E+01	1282	1367
0.0000 I	0.00000E+00	0.10000E+02	0.10184E+02 ✓	0.10030E+02	0.91118E+01	1282	1304
0.0000 I	0.00000E+00	0.10000E+02	0.10187E+02 ✓	0.10533E+02	0.97918E+01	1282	1302
0.0000 I	0.00000E+00	0.10000E+02	0.97493E+01	0.10920E+02	0.91655E+01	1282	1241
0.0000 I	0.00000E+00	0.10000E+02	0.10523E+02	0.10201E+02	0.10380E+02	1282	1364
0.0000 I	0.00000E+00	0.10000E+02	0.10455E+02	0.10515E+02	0.10607E+02	1282	1301
0.0000 I	0.00000E+00	0.10000E+02	0.10334E+02 ✓	0.10047E+02	0.10120E+02	1282	1302
0.0000 I	0.00000E+00	0.10000E+02	0.91625E+01	0.10143E+02	0.89547E+01	1282	1178
0.0000 I	0.00000E+00	0.10000E+02	0.99315E+01	0.10003E+02	0.10848E+02	1282	1237
0.0000 I	0.00000E+00	0.10000E+02	0.10336E+02 ✓	0.98425E+01	0.95713E+01	1282	1303
0.0000 I	0.00000E+00	0.10000E+02	0.10645E+02	0.10345E+02	0.11313E+02	1282	1362
0.0000 I	0.00000E+00	0.10000E+02	0.92917E+01	0.96469E+01	0.11340E+02	1282	1173
0.0000 I	0.00000E+00	0.10000E+02	0.10604E+02	0.10288E+02	0.97223E+01	1282	1366
0.0000 I	0.00000E+00	0.10000E+02	0.10634E+02	0.10740E+02	0.85350E+01	1282	1368
0.0000 I	0.00000E+00	0.10000E+02	0.96133E+01	0.10516E+02	0.10277E+02	1282	1238

The corresponding z-coordinates are highlighted.

The output file generated by StatSpatial V1.0 is shown below. (the section contains non-zero values) (Test StatSpatial.out)

Z	
0.12250E+02	0
0.12150E+02	0
0.12050E+02	0
0.11950E+02	0
0.11850E+02	0
0.11750E+02	0
0.11650E+02	0
0.11550E+02	0
0.11450E+02	0
0.11350E+02	1
0.11250E+02	0
0.11150E+02	0
0.11050E+02	0
0.10950E+02	0
0.10850E+02	0
0.10750E+02	0
0.10650E+02	0
0.10550E+02	0
0.10450E+02	0
0.10350E+02	0
0.10250E+02	1
0.10150E+02	1
0.10050E+02	0
0.99500E+01	0
0.98500E+01	1
0.97500E+01	0
0.96500E+01	1
0.95500E+01	0
0.94500E+01	0
0.93500E+01	0
0.92500E+01	0
0.91500E+01	1
0.90500E+01	0
0.89500E+01	0
0.88500E+01	0
0.87500E+01	0
0.86500E+01	0
0.85500E+01	1
0.84500E+01	0
0.83500E+01	0
0.82500E+01	0
0.81500E+01	0
0.80500E+01	0
0.79500E+01	0
0.78500E+01	0
0.77500E+01	0
0.76500E+01	0
0.75500E+01	0

# of particles in Y = 10.15 to 10.35 range

LP 3/6/00

example on p. 130

LP 3/6/00

A throughly ~~check~~ check is done and the

LP 2/3/2000

SIGNATURE _____ DATE _____ 19 _____

READ AND UNDERSTOOD _____ DATE _____ 19 _____

PROJECT NAME _____ NOTEBOOK NO. _____

results are correct and accurate. For example the point on the output file, 0.11350E+02, has a window [11.30, 11.40]. Only one particle at Z = 0.11348E+02 in the input file is located within its window. Therefore, its particle count is 1, which is shown in the output file.

Note that, all values in this case are dimensionless.

LP

SIGNATURE _____ DATE _____ 19__

READ AND UNDERSTOOD _____ DATE _____ 19__

YMP-LBNL
REVIEW RECORD

1. QA: L
2. Page 1 of 1

3. Originator: Lehua Pan

4. Document Title: Documentation for Routine **StatSpatial v1.0** (Option 1 per AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1)

5. Document Number: N/A 6. Revision/Mod.: N/A 7. Draft: N/A

8. Governing Procedure Number: AP-SI.1Q 9. Revision/Mod: 2/4

REVIEW CRITERIA: YMP-LBNL-QIP-6.1, Attch. 5, p. 18 Routine Review Criteria

10. ☒ Standard Review Criteria (One time use Option 1) (Taken from Attachment 5)

11. ☐ Specific Review Criteria:
☒ Source: AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1.1 (One time use routine)

12. Comment Documentation:
☐ Comment Sheets
☐ Review Copy Mark-up
☐ Attached:

☒ Scientific notebook/data associated with this review as noted on Attachment 3

13. YMP-LBNL Project Manager (PM): Gudmundur S. Bodvarsson

14. Reviewer	Org./Discipline	Review Criteria	Reviewer	Org./Discipline	Review Criteria
R.F. Hedegaard	LBNL/Hydrogeologist	Technical			

COMMENTS DUE:

15. Due Date: 2/18/00

16. Originator/Review Coordinator: Lehua Pan
Print Name

REVIEW BY:

17. Randall F. Hedegaard
Print Name

18. [signature] 1MAR00
Signature Date

19. Mandatory Comments: ☐ Yes ☒ No

CONCURRENCE:

21. Document Draft No: NA Date: -

22. Reviewer: [signature] 2MAR00
Signature Date

23. PM: [signature] 3/3/00
Signature Date

ORIGINATOR/REVIEW COORDINATOR (After response completed):

20. Lehua Pan [signature] 3/3/00
Print Name/Signature Date

DISPUTE RESOLUTION: (if applicable)

24. PM:
Signature Date

YMP-LBNL
COMMENT SHEET

QA: L

1. Document Title: **Routine Documentation for StatSpatial v1.0**			2. Page 1 of 1
3. Document No. **N/A**	4. Revision/ Change/Mod: **N/A**	5. Draft **N/A**	6. ☒ Q ☐ NQ
7. Reviewer: **Randall F. Hedegaard**			

8. NO. CODE	9. SECT./PARA./P#	10. COMMENT	11. RESPONSE	12. ACCEPT
		--NO COMMENTS— The documentation for this routine was reviewed and it was found to meet the requirements of AP-SI.1Q/Rev. 2/ICN4. The test case was checked by both hand calculation and by running the code as needed to fully check the test case. The test case fully checks the routine for the input specified and proves that the routine produces acceptable results.		

YMP-LBNL
APPLICABLE REFERENCE INFORMATION

Document No.and Title: Routine Documentation for **StatSpatial v1.0** per Option 1, AP-SI.1Q/Rev.2/ICN4, Sec. 5.5.1

Date of Document (or revision, draft revision number, as applicable):
NA

Pertinent sections of scientific notebook(s) or other backup documents and/or data DTN# are identified below, supporting the document which is the subject of this review. These documents/data shall be included in the scope of this review.

Document(s) Title/Data	Relevant Sections/Pages
YMP-LBNL-GSB-LP-3 Scientific Notebook	pp. 76-79, 128-130

STANDARD REVIEW CRITERIA

Routine/Macro Review Criteria, Option 1

NOTE: Where a checklist item does not apply to the software product, check "N/A".

	Yes	No	N/A	
R/M-1	X			**The information given below is to be documented in the technical product, in which the routine/macro is used to support. Does the routine/macro include:** Name of routine/macro with version/Operating System/hardware environment
R/M-2	X			Name of commercial software used to write the routine/macros with version/Operating System/hardware used to develop it
R/M-3	X			Test Plan • Explanation whether this is a routine or macro and a description of what it does • The source code (this section shall include equations or algorithms form software setup (Labview, Excel, etc.) • Description of test(s) to be performed (be specific) • Specified range of input values to be used and why the range is valid
R/M-4	X			Test Results • Output from test (explain difference between input range used and possible input) • Description of how the testing shows that the results are correct for the specified input • List of limitations or assumptions to this test case (s) and code in general • Electronic files identified by name and location (included if necessary to perform the tests)
R/M-5	X			Supporting Information. Include background information, such as revision to a previous routine or macro or explanation of the steps performed to run the software. Include listing of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

ExtractFlow v.1.0
Routine/Macro Documentation Form

The following information can be included in the scientific notebook. Attach and reference notebook pages and diskettes with files as needed when submitting routine/macro to records.

1. Name of routine/macro with version/OS/hardware environment:
   ExtractFlow V1.0 (routine) / Windows 98/PC

2. Name of commercial software with version/OS/hardware used to develop routine/macro:
   Digital Visual FORTRAN 5.0 (Fortran 90)/Windows 98/PC

3. **Test Plan.**

   - Explain whether this is a routine or macro and describe what it does:
     This routine is used to create a DCPT V1.0 readable file of the flow rates per connections from the TOUGH2 output file.

   - Source code: (including equations or algorithms from software setup (LabView, Excel, etc.):
     p. 92 of YMP-LBNL-GSB-LP-3 (attached)

   - Description of test(s) to be performed (be specific):
     The routine will first search for the keyword "FLO(LIQ)" from the TOUGH2 V1.4 file (paul_trl.out) and then read the subsequent information and write it to the output file (PAU1.flow) until the end of the file. The format (ordering within a row) of the data in the output file is exactly the same as those in the input file. No calculation is involved in this routine. To verify the reformatting, a representative sampling of the input and output files will be used.

   - Specify the range of input values to be used and why the range is valid
     The input data are lines from the output file of TOUGH2 V1.4 (pau1_tr1.out) printed on p. 127, YMP-LBNL-GSB-LP-3. that are used in the related analysis. The test case is actually the single use case that the routine is designed for. Thus, this test case input range is deemed valid.

4. **Test Results.**

   - Output from test (explain difference between input range used and possible input)
     The output are several lines from the file "PAU1.flow" printed on pp. 127 of YMP-LBNL-GSB-LP-3. To verify the reformatting, a representative sampling of the input and output files was used.

   - Description of how the testing shows that the results are correct for the specified input.
     Because the routine performs reformatting of data only, a spot-check for consistency between the data within the input file and the output file is sufficient to justify that the routine works properly. The test was successful because the routine successfully ran without error messages and the data in the new "PAU1.flow" are numerically identical to their counterparts in the input file. No calculation is involved in the routine. Therefore, the test case and routine are acceptable.

   - List limitations or assumptions to this test case and code in general
     The routine was tested using an input data set that is actually used in the related analysis. The routine is only valid for the output file of TOUGH2 V1.4. The output

file can only be used by DCPT V1.0. The routine is considered as a single-user routine.

- Electronic files identified by name and location (include disc if necessary)
  The routine, test files and its description can be found on pp. 92 and 126-127 from YMP-LBNL-GSB-LP-3.

5. Supporting Information. Include background information, such as revision to a previous routine or macro, or explanation of the steps performed to run the software. Include listings of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

See attached pages for technical review forms, referenced scientific notebook pages and other supporting documentation

Note: All relevant S/N pages are included in this package. In some instances, the included S/N pages cross-reference other pages that are not included here because these were not essential to the documentation of this routine.

MAINTAIN PAGES IN THIS ORDER:

1. **This 2-page routine documentation form.**
2. **pp. 92, 126-127 from YMP-LBNL-GSB-LP-3**
3. **Review forms**

the output file. No calculation is involved.
— A spot checking is provided below.
(uzm_trz.dat) LP 12/23/99
The data in the input file (highlighted) are

```
tswM1    2 2510.    0.5300E-010.6321E-160.6321E-160.6321E-16 1.000     1000.
                          0.7200      0.7000
     0.00       0.00        0.0      1.0e-3  1.60e-10
   7      0.3033     0.2200      1.000                   1.000
   7      0.3033     0.2200     0.3612E-040.1000E+11 1.000
tswM2    2 2550.    0.1570     0.5825E-150.5825E-150.5825E-15 1.000     1000.
                          0.7200      0.7000
     0.00       0.00        0.0      1.0e-3  1.60e-10
   7      0.3327     0.7000E-01 1.000                   1.000
   7      0.3327     0.7000E-010.3607E-040.1000E+11 1.000
tswM3    2 2510.    0.1540     0.3083E-160.3083E-160.3083E-16 1.000     1000.
                          0.7200      0.7000
     0.00       0.00        0.0      1.0e-3  1.60e-10
   7      0.2978     0.1200      1.000                   1.000
   7      0.2978     0.1200     0.2125E-040.1000E+11 1.000
```

LP 3/6/00

The corresponding data in the output file (UZ99.Kd) are (highlighted):

```
12  1.000E-03    2510.00  0.000E+00    2510.00
13  1.000E-03    2550.00  0.000E+00    2550.00
14  1.000E-03    2510.00  0.000E+00    2510.00
```

The data are identical numerically.

Test of ExtractFlow V1.0

For full description of the code, see page 91-92. The program ~~is in~~ was running LP 12/23/99 under Digital Visual Fortran developer environment on ~~a~~ the PC (LBL/DOE 631 7820). LP 12/23/99

The input file is "paul_tr1.out" Located in "E:\ParticleTrack\Cal99\". The output file

SIGNATURE ______ DATE ______ 19 __
READ AND UNDERSTOOD ______ DATE ______ 19 __

PROJECT NAME _____ NOTEBOOK NO. _____

Is "PAul.flow" located in "E:\Particle Track"?

The program only transfers data from the input file to the output file. No calculation is involved. A spot checking is provided below:

LP 12/23/99

The data in the input file "paul_tri.dat" are (highlighted): (first four)

Upper Inf. #1 perched water conceptual model ysw 5/29/99

KCYC = 365 - ITER = 2 - TIME = 0.31558E+14

ELEM1	ELEM2	INDEX	FLO(LIQ) (KG/S)	VEL (M/S)	(FLO_X3) (KG/S)	(FLO_CUM) (KG)
Fea 1	Fda 1	1	0.36099464E-01	0.40874399E-09	0.24989958E-70	0.28429005E-54
Mea 1	Mda 1	2	0.49648209E-02	0.56215223E-10	0.23855256E-70	0.14511515E-54
Fda 1	Fca 1	3	0.40995825E-01	0.46418542E-09	0.25242851E-70	0.31345637E-54
Mda 1	Mca 1	4	0.95955948E-04	0.10864814E-11	0.14010939E-71	0.18240193E-55
Fca 1	Fba 1	5	0.40890726E-01	0.46299541E-09	0.45271439E-71	0.52721224E-55

The corresponding data in the output file "PAul.flow" are (highlighted):

ELEM1	ELEM2	INDEX	FLO(LIQ) (KG/S)	VEL (M/S)	(FLO_X3) (KG/S)	(FLO_CUM) (KG)
Fea 1	Fda 1	1	0.36099464E-01	0.40874399E-09	0.24989958E-70	0.28429005E-54
Mea 1	Mda 1	2	0.49648209E-02	0.56215223E-10	0.23855256E-70	0.14511515E-54
Fda 1	Fca 1	3	0.40995825E-01	0.46418542E-09	0.25242851E-70	0.31345637E-54
Mda 1	Mca 1	4	0.95955948E-04	0.10864814E-11	0.14010939E-71	0.18240193E-55

They are identical.

LP 3/6/00

LP

SIGNATURE _____ DATE _____ 19 ____

READ AND UNDERSTOOD _____ DATE _____ 19 ____

ExtractFlow V1.0

Lehua Pan

Creator

Lehua Pan 3/6/00

```
      Program ExtractFlow
*     to extract the flow per connection data
      implicit none

      integer i
      character Keyword*100
      Character*50 T2outFile,FlowFile,InputDir,OutputDir
      InputDir="E:\ParticleTrack\Cal99\"
      OutPutDir="E:\ParticleTrack\"
      T2OUTFile=trim(InputDir) // "pau1_tr1.out"

      Flowfile=trim(OUTputDir) // "PAu1.flow"


      open(11,file=T2OutFile,status='old', err=40)

      open (16,file=Flowfile)


cc    Get rid of garbbage
      do while (keyword(34:42).ne.'FLO(LIQ)')
         read(11,'(a100)',err=33) keyword
      enddo
      write (*,*) keyword(34:42)
      pause
c
      do i=1,5
         write (16,'(a100)') keyword
      enddo
      i=0
      do while (keyword(4:4) .eq. keyword(13:13)
     1    .or. keyword(13:13) .eq. "T" .or. keyword(4:4) .eq. "B")
!     do while (keyword .ne. "")
         read (11,'(a100)',err=33) keyword
         write (16,'(a100)') keyword
      i=i+1
      if ( mod(i, 100) .eq. 0) then
         write (*,*) 'Count=',i
      endif


      !pause
      enddo


      close (16)
      close (11)

20      stop "!done"
33    stop "!error in reading TOUGH output file!!"

40    stop "!No TOUGH output file!"


      end
```

1

LP

SIGNATURE _____ DATE _____ 19

READ AND UNDERSTOOD _____ DATE _____ 19

YMP-LBNL
REVIEW RECORD

1. QA: L
2. Page 1 of 1

3. Originator: Lehua Pan

4. Document Title: Documentation for Routine **ExtractFlow v1.0** (Option 1 per AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1)

5. Document Number: N/A 6. Revision/Mod.: N/A 7. Draft: N/A

8. Governing Procedure Number: AP-SI.1Q 9. Revision/Mod: 2/4

REVIEW CRITERIA

10. ☒ Standard Review Criteria — YMP-LBNL-QIP-6.1, Attch. 5, p. 18 Routine Review Criteria (One time use Option 1) (Taken from Attachment 5)

11. ☐ Specific Review Criteria:
☒ Source: AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1.1 (One time use routine)

12. Comment Documentation:
☐ Comment Sheets
☐ Review Copy Mark-up
☐ Attached:
☒ Scientific notebook/data associated with this review as noted on Attachment 3

13. YMP-LBNL Project Manager (PM): Gudmundur S. Bodvarsson

14. Reviewer	Org./Discipline	Review Criteria	Reviewer	Org./Discipline	Review Criteria
R.F. Hedegaard	LBNL/Hydrogeologist	Technical			

COMMENTS DUE:

15. Due Date: 2/18/00

16. Originator/Review Coordinator: Lehua Pan
Print Name

REVIEW BY:

17. Randall F. Hedegaard
Print Name

18. [signature] 1MAR00
Signature Date

19. Mandatory Comments: ☐ Yes ☒ No

CONCURRENCE:

21. Document Draft No: NA Date: ←

22. Reviewer: [signature] 2MAR00
Signature Date

23. PM: [signature] 3/3/00
Signature Date

ORIGINATOR/REVIEW COORDINATOR (After response completed):

20. Lehua Pan Lehua Pan 3/3/00
Print Name/Signature Date

DISPUTE RESOLUTION: (if applicable)

24. PM:
Signature Date

YMP-LBNL
COMMENT SHEET

QA: L

1. Document Title:			2. Page 1 of 1
Routine Documentation for ExtractFlow v1.0			
3 . Document No.	4. Revision/ Change/Mod:	5. Draft	6. ☒ Q ☐ NQ
N/A	**N/A**	**N/A**	
7. Reviewer:			
Randall F. Hedegaard			

8. NO. CODE	9. SECT./PARA./P#	10. COMMENT	11. RESPONSE	12. ACCEPT
		--NO COMMENTS— The documentation for this routine was reviewed and it was found to meet the requirements of AP-SI.1Q/Rev. 2/ICN4. The test case was checked by both hand calculation and by running the code as needed to fully check the test case. The test case fully checks the routine for the input specified and proves that the routine produces acceptable results.		

YMP-LBNL
APPLICABLE REFERENCE INFORMATION

Document No.and Title: Routine Documentation for **ExtractFlow v1.0** per Option 1, AP-SI.1Q/Rev.2/ICN4, Sec. 5.5.1

Date of Document (or revision, draft revision number, as applicable):
NA

Pertinent sections of scientific notebook(s) or other backup documents and/or data DTN# are identified below, supporting the document which is the subject of this review. These documents/data shall be included in the scope of this review.

Document(s) Title/Data	Relevant Sections/Pages
YMP-LBNL-GSB-LP-3 Scientific Notebook	pp. 92, 126-127

STANDARD REVIEW CRITERIA

Routine/Macro Review Criteria, Option 1

NOTE: Where a checklist item does not apply to the software product, check "N/A".

	Yes	No	N/A	
R/M-1	X			**The information given below is to be documented in the technical product, in which the routine/macro is used to support. Does the routine/macro include:** Name of routine/macro with version/Operating System/hardware environment
R/M-2	X			Name of commercial software used to write the routine/macros with version/Operating System/hardware used to develop it
R/M-3	X			Test Plan • Explanation whether this is a routine or macro and a description of what it does • The source code (this section shall include equations or algorithms form software setup (Labview, Excel, etc.) • Description of test(s) to be performed (be specific) • Specified range of input values to be used and why the range is valid
R/M-4	X			Test Results • Output from test (explain difference between input range used and possible input) • Description of how the testing shows that the results are correct for the specified input • List of limitations or assumptions to this test case (s) and code in general • Electronic files identified by name and location (included if necessary to perform the tests)
R/M-5	X			Supporting Information. Include background information, such as revision to a previous routine or macro or explanation of the steps performed to run the software. Include listing of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

ExBT v.1.0
Routine/Macro Documentation Form

The following information can be included in the scientific notebook. Attach and reference notebook pages and diskettes with files as needed when submitting routine/macro to records.

1. Name of routine/macro with version/OS/hardware environment:
   ExBT V1.0 (routine) / Windows 98/PC

2. Name of commercial software with version/OS/hardware used to develop routine/macro:
   Digital Visual FORTRAN 5.0 (Fortran 90)/Windows 98/PC

3. **Test Plan.**

   - Explain whether this is a routine or macro and describe what it does:
     This routine is used to extract the breakthrough curve (mass vs. time) from the T2R3D V1.4 output file.

   - Source code: (including equations or algorithms from software setup (LabView, Excel, etc.):
     p. 93 of YMP-LBNL-GSB-LP-3 (attached)

   - Description of test(s) to be performed (be specific):
     The routine will first search the keyword "ITERATING" from the T2R3D V1.4 file (uzm_trz.out) and then read the time and the corresponding mass and write them to the output file (UZ99.NP). The loop will keep going until reaching the end of the file. The first column of data in the output file (UZ99.NP) is time (years) while the second column is the corresponding mass leaving the domain at the bottom.
     To verify the reformatting, a representative sampling of the input and output files will be used.

   - Specify the range of input values to be used and why the range is valid
     The test case is actually the single use case that the routine is designed for. Thus, this test case input range is deemed valid.

4. **Test Results.**

   - Output from test (explain difference between input range used and possible input)
     The output are several lines from the file "UZ99.NP" printed on pp. 96-97 of YMP-LBNL-GSB-LP-3. To verify the reformatting, a representative sampling of the input and output files was used.

   - Description of how the testing shows that the results are correct for the specified input.
     The reformatting was successful because the comparison of the output and input files on p. 94-97, YMP_LBNL-GSB-LP-3 shows identical numbers. Also, the test was acceptable because the routine successfully ran without error messages and the new "UZ99.NP" was successfully imported into DCPT V1.0 without errors. No calculation is involved in the routine. Therefore, the test case and routine are acceptable.

   - List limitations or assumptions to this test case and code in general
     The routine was tested using an input data set that is actually used in the related analysis. However, the routine is only valid for the output file of T2R3D V1.4. The routine is considered as a single-use routine.

- Electronic files identified by name and location (include disc if necessary)
  The routine, test files and its description can be found on pp. 93-97 from YMP-LBNL-GSB-LP-3.

5. Supporting Information. Include background information, such as revision to a previous routine or macro, or explanation of the steps performed to run the software. Include listings of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.

See attached pages for technical review forms, referenced scientific notebook pages and other supporting documentation

See attached pages for technical review forms, referenced scientific notebook pages and other supporting documentation

Note: All relevant S/N pages are included in this package. In some instances, the included S/N pages cross-reference other pages that are not included here because these were not essential to the documentation of this routine.

MAINTAIN PAGES IN THIS ORDER:

1. **This 2-page routine documentation form.**
2. **pp. 93-97, from YMP-LBNL-GSB-LP-3**
3. **Review forms**

9/1/99 Extract Breakthrough curve from T2R3D output

ExBT V1.0 Lehua Pan

Lehua Pan created 3/6/00

```
c This program Extract time series of cumulative mass flow out of the domain
!     from TOUGH2 output file for plotting BT curver
!        written by Lehua Pan 7/6/99
      program ExBT
      implicit none

      double precision T,Mass
      character Keyword*19
      Character*50 T2outFile,BTFile,InputDir,OutputDir
      InputDir="E:\ParticleTrack\Cal99\"
      OutPutDir="E:\ParticleTrack\"
      T2OUTFile=trim(InputDir) // "uzm_tr2.out"

      BTfile=trim(OUTputDir) // "UZ99.np"


      open(11,file=T2OutFile,status='old', err=40)

      open (16,file=BTfile)


cc    Get rid of garbbage
      do while (keyword(5:13).ne.'ITERATING')
         read(11,'(a19)',err=33) keyword
      enddo
      write (*,*) keyword(5:13)
      pause
c
      do while (keyword(5:13) .eq. 'ITERATING')
          read (11,'(a19, e12.6, 35x, e12.6)',err=30) keyword,t,mass
          write (16,'(2e15.6)') t,mass
      write (*,*) keyword,t,mass
10        do while (keyword(5:13).ne.'ITERATING')
              read(11,'(a19)',end=20) keyword
!             write (*,*) keyword
          enddo


      !pause
      enddo


      close (16)
      close (11)

20      stop "!done"

30    goto 10
33    stop "!error in reading TOUGH output file!!"

40    stop "!No TOUGH output file!"


      end
```

1

SIGNATURE _____ DATE _____ 19 _____

READ AND UNDERSTOOD _____ DATE _____ 19 _____

This program is used to extract the cumulative mass out from the domain into water table and the corresponding time in the T2R3D output file.

It first searches the keyword "ITERATING" and then read t and mass, and writes to the new output file, and so on. LP 9/1/99

Verification the Code:

The following is the first part of the related section of "uzm_tr2.out": LP 9/1/99

Test case input:

```
...ITERATING...  AT [   1,  1] --- DELTEX = 0.100000E+01   MAX. RES. = 0.123875E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.316881E-07 BOTTOM CUM. TRACER FLUX (kg) =    0.353060E-57 # of Connections =  2868
Fsc 5   (   1,  2) ST = 0.100000E+01 DT = 0.100000E+01 DX1= -.533751E-06

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   2,  1] --- DELTEX = 0.130000E+01   MAX. RES. = 0.160957E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.728826E-07 BOTTOM CUM. TRACER FLUX (kg) =    0.842319E-56 # of Connections =  2868
Fqj35   (   2,  2) ST = 0.230000E+01 DT = 0.130000E+01 DX1= -.523842E-06

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   3,  1] --- DELTEX = 0.169000E+01   MAX. RES. = 0.208334E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.126435E-06 BOTTOM CUM. TRACER FLUX (kg) =    0.117247E-54 # of Connections =  2868
Fsj40   (   3,  2) ST = 0.399000E+01 DT = 0.169000E+01 DX1= -.724156E-06

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   4,  1] --- DELTEX = 0.219700E+01   MAX. RES. = 0.270833E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.196054E-06 BOTTOM CUM. TRACER FLUX (kg) =    0.125238E-53 # of Connections =  2868
Fpj52   (   4,  2) ST = 0.618700E+01 DT = 0.219700E+01 DX1= -.988631E-06

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   5,  1] --- DELTEX = 0.285610E+01   MAX. RES. = 0.352082E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.286559E-06 BOTTOM CUM. TRACER FLUX (kg) =    0.114176E-52 # of Connections =  2868
Flj57   (   5,  2) ST = 0.904310E+01 DT = 0.285610E+01 DX1= -.199185E-05

  average inner-iteration/solve    =    1.00000000000000

WRITE FILE *SAVE* AFTER    4 TIME STEPS  ---  THE TIME IS 0.904310E+01 SECONDS

...ITERATING...  AT [   6,  1] --- DELTEX = 0.371293E+01   MAX. RES. = 0.457705E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.404214E-06 BOTTOM CUM. TRACER FLUX (kg) =    0.938436E-52 # of Connections =  2868
Fkj54   (   6,  2) ST = 0.127560E+02 DT = 0.371293E+01 DX1= -.220397E-05

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   7,  1] --- DELTEX = 0.482681E+01   MAX. RES. = 0.595013E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.557167E-06 BOTTOM CUM. TRACER FLUX (kg) =    0.718207E-51 # of Connections =  2868
Fqj35   (   7,  2) ST = 0.175828E+02 DT = 0.482681E+01 DX1= -.194498E-05

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   8,  1] --- DELTEX = 0.627485E+01   MAX. RES. = 0.773512E-04  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.756005E-06 BOTTOM CUM. TRACER FLUX (kg) =    0.522483E-50 # of Connections =  2868
Fsj40   (   8,  2) ST = 0.238577E+02 DT = 0.627485E+01 DX1= -.268872E-05

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [   9,  1] --- DELTEX = 0.815731E+01   MAX. RES. = 0.100556E-03  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.101449E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.366355E-49 # of Connections =  2868
Flj57   (   9,  2) ST = 0.320150E+02 DT = 0.815731E+01 DX1= -.568883E-05

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  10,  1] --- DELTEX = 0.106045E+02   MAX. RES. = 0.130721E-03  AT ELEMENT Fmj57     EQUATION   1
 TIME (years) =  0.135053E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.250002E-48 # of Connections =  2868
Pkj54   (  10,  2) ST = 0.426195E+02 DT = 0.106045E+02 DX1= -.629464E-05

  average inner-iteration/solve    =    1.00000000000000
```

```
WRITE FILE *SAVE* AFTER    9 TIME STEPS  ---  THE TIME IS 0.426195E+02 SECONDS

...ITERATING...  AT [  11,  1] --- DELTEX = 0.137858E+02   MAX. RES. = 0.169935E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.178738E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.167185E-47 # of Connections =  2868
Flj57   (  11,  2) ST = 0.564053E+02 DT = 0.137858E+02 DX1= -.961395E-05

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  12,  1] --- DELTEX = 0.179216E+02   MAX. RES. = 0.220911E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.235528E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.110117E-46 # of Connections =  2868
Flj57   (  12,  2) ST = 0.743270E+02 DT = 0.179216E+02 DX1= -.124980E-04

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  13,  1] --- DELTEX = 0.232981E+02   MAX. RES. = 0.287177E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.309355E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.717007E-46 # of Connections =  2868
Fpj52   (  13,  2) ST = 0.976250E+02 DT = 0.232981E+02 DX1= -.104835E-04

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  14,  1] --- DELTEX = 0.302875E+02   MAX. RES. = 0.373318E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.405330E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.462818E-45 # of Connections =  2868
Fqj35   (  14,  2) ST = 0.127913E+03 DT = 0.302875E+02 DX1= -.122039E-04

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  15,  1] --- DELTEX = 0.393738E+02   MAX. RES. = 0.485293E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.530098E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.296768E-44 # of Connections =  2868
Fqj35   (  15,  2) ST = 0.167286E+03 DT = 0.393738E+02 DX1= -.158648E-04

  average inner-iteration/solve    =    1.00000000000000

WRITE FILE *SAVE* AFTER   14 TIME STEPS  ---  THE TIME IS 0.167286E+03 SECONDS

...ITERATING...  AT [  16,  1] --- DELTEX = 0.511859E+02   MAX. RES. = 0.630846E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.692297E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.189332E-43 # of Connections =  2868
Fkj54   (  16,  2) ST = 0.218472E+03 DT = 0.511859E+02 DX1= -.303799E-04

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  17,  1] --- DELTEX = 0.665417E+02   MAX. RES. = 0.820042E-03  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.903154E-05 BOTTOM CUM. TRACER FLUX (kg) =    0.120323E-42 # of Connections =  2868
Flj57   (  17,  2) ST = 0.285014E+03 DT = 0.665417E+02 DX1= -.463973E-04

  average inner-iteration/solve    =    1.00000000000000
...ITERATING...  AT [  18,  1] --- DELTEX = 0.865042E+02   MAX. RES. = 0.106596E-02  AT ELEMENT Fmj57     EQUATION  1
 TIME (years) =   0.117727E-04 BOTTOM CUM. TRACER FLUX (kg) =    0.762397E-42 # of Connections =  2868
Fsj40   (  18,  2) ST = 0.371518E+03 DT = 0.865042E+02 DX1= -.370607E-04

  average inner-iteration/solve    =    1.00000000000000
```


The following is the file "UZ99.NP" saved by this program. Check the first three data, they are: LP 9/1/89

~~Time = 0.728826E-07~~

0.316881E-07 0.353060E-57

0.728826E-07 0.842319E-56

0.126435E-06 0.117247E-54 in "Uzm_tr2.out"

0.316881E-07 0.353060E-57
0.728826E-07 0.842319E-56
0.126435E-06 0.117247E-54 in "UZ99.NP"

SIGNATURE _____ DATE _____ 19 ____

READ AND UNDERSTOOD _____ DATE _____ 19 ____

U299.NP

Time Flux (Kg)

0.316881E-07	0.353060E-57
0.728826E-07	0.842319E-56
0.126435E-06	0.117247E-54
0.196054E-06	0.125238E-53
0.286559E-06	0.114176E-52
0.404214E-06	0.938436E-52
0.557167E-06	0.718207E-51
0.756005E-06	0.522483E-50
0.101449E-05	0.366355E-49
0.135053E-05	0.250002E-48
0.178738E-05	0.167185E-47
0.235528E-05	0.110117E-46
0.309355E-05	0.717007E-46
0.405330E-05	0.462818E-45
0.530098E-05	0.296768E-44
0.692297E-05	0.189332E-43
0.903154E-05	0.120323E-42
0.117727E-04	0.762397E-42
0.153362E-04	0.481974E-41
0.199687E-04	0.304161E-40
0.259910E-04	0.191687E-39
0.338201E-04	0.120677E-38
0.439978E-04	0.759098E-38
0.572288E-04	0.477186E-37
0.744291E-04	0.299811E-36
0.967895E-04	0.188286E-35
0.125858E-03	0.118202E-34
0.163647E-03	0.741786E-34
0.212773E-03	0.465357E-33
0.276637E-03	0.291837E-32
0.359659E-03	0.182947E-31
0.467589E-03	0.114633E-30
0.607897E-03	0.717900E-30
0.790298E-03	0.449340E-29
0.102742E-02	0.281297E-28
0.133568E-02	0.1773349E-27
0.173641E-02	0.118665E-26
0.225737E-02	0.1124459E-25
0.293461E-02	0.251223E-24
0.381502E-02	0.987436E-23
0.495956E-02	0.430607E-21
0.644746E-02	0.184274E-19
0.838173E-02	0.753019E-18
0.108963E-01	0.289558E-16
0.141652E-01	0.103217E-14
0.184148E-01	0.335310E-13
0.239392E-01	0.973800E-12
0.311210E-01	0.247586E-10
0.404574E-01	0.539175E-09
0.525946E-01	0.984263E-08
0.683731E-01	0.147651E-06
0.888850E-01	0.179007E-05
0.115551E+00	0.173295E-04
0.150216E+00	0.133117E-03
0.195281E+00	0.811466E-03
0.253865E+00	0.395805E-02
0.330024E+00	0.158823E-01
0.429031E+00	0.585541E-01
0.557741E+00	0.271119E+00
0.725063E+00	0.183339E+01
0.942582E+00	0.127634E+02
0.122536E+01	0.720301E+02
0.159296E+01	0.316693E+03
0.207085E+01	0.111276E+04
0.269211E+01	0.324120E+04
0.349974E+01	0.801624E+04
0.454966E+01	0.169189E+05
0.591456E+01	0.303959E+05
0.768893E+01	0.467288E+05
0.999561E+01	0.626994E+05
0.129943E+02	0.757795E+05
0.168926E+02	0.853830E+05
0.219604E+02	0.921751E+05
0.285485E+02	0.969892E+05
0.371130E+02	0.100509E+06

1

SIGNATURE _____ DATE _____ 19
READ AND UNDERSTOOD _____ DATE _____ 19

0.482469E+02	0.103256E+06
0.627210E+02	0.105501E+06
0.815373E+02	0.107332E+06
0.105998E+03	0.108864E+06
0.137798E+03	0.110313E+06
0.179137E+03	0.111868E+06
0.232879E+03	0.113588E+06
0.302742E+03	0.115434E+06
0.393565E+03	0.117392E+06
0.511634E+03	0.119539E+06
0.665125E+03	0.122014E+06
0.864662E+03	0.124966E+06
0.112406E+04	0.128540E+06
0.146128E+04	0.132890E+06
0.189966E+04	0.138217E+06
0.246956E+04	0.144811E+06
0.321043E+04	0.153109E+06
0.417356E+04	0.163778E+06
0.542563E+04	0.177834E+06
0.705331E+04	0.196822E+06
0.916931E+04	0.223002E+06
0.119201E+05	0.259425E+06
0.150889E+05	0.303765E+06
0.182577E+05	0.349531E+06
0.214265E+05	0.395829E+06
0.245953E+05	0.441925E+06
0.277641E+05	0.487271E+06
0.309330E+05	0.531499E+06
0.341018E+05	0.574383E+06
0.372706E+05	0.615815E+06
0.404394E+05	0.655763E+06
0.436082E+05	0.694253E+06
0.467770E+05	0.731345E+06
0.499458E+05	0.767115E+06
0.531146E+05	0.801651E+06
0.562834E+05	0.835041E+06
0.594522E+05	0.867371E+06
0.626210E+05	0.898721E+06
0.657899E+05	0.929164E+06
0.689587E+05	0.958768E+06
0.721275E+05	0.987591E+06
0.752963E+05	0.101569E+07
0.784651E+05	0.104311E+07
0.816339E+05	0.106989E+07
0.848027E+05	0.109607E+07
0.879715E+05	0.112169E+07
0.911403E+05	0.114677E+07
0.943091E+05	0.117134E+07
0.974779E+05	0.119542E+07
0.100647E+06	0.121905E+07
0.103816E+06	0.124223E+07
0.106984E+06	0.126499E+07
0.110153E+06	0.128734E+07
0.113322E+06	0.130929E+07
0.116491E+06	0.133087E+07
0.119660E+06	0.135208E+07
0.122828E+06	0.137294E+07
0.125997E+06	0.139346E+07
0.129166E+06	0.141364E+07
0.132335E+06	0.143350E+07
0.135504E+06	0.145306E+07
0.138672E+06	0.147231E+07
0.141841E+06	0.149126E+07
0.145010E+06	0.150993E+07
0.148179E+06	0.152832E+07
0.151348E+06	0.154644E+07
0.154516E+06	0.156429E+07
0.157685E+06	0.158189E+07
0.160854E+06	0.159923E+07
0.164023E+06	0.161633E+07
0.167192E+06	0.163320E+07
0.170361E+06	0.164982E+07
0.173529E+06	0.166622E+07
0.176698E+06	0.168240E+07
0.179867E+06	0.169836E+07

SIGNATURE _____ DATE _____ 19 _____
READ AND UNDERSTOOD _____ DATE _____ 19 _____

YMP-LBNL
REVIEW RECORD

1. QA: L
2. Page 1 of 1

3. Originator: Lehua Pan

4. Document Title: Documentation for Routine **ExBT v1.0** (Option 1 per AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1)

5. Document Number: N/A 6. Revision/Mod.: N/A 7. Draft: N/A

8. Governing Procedure Number: AP-SI.1Q 9. Revision/Mod: 2/4

REVIEW CRITERIA — YMP-LBNL-QIP-6.1, Attch. 5, p. 18 Routine Review Criteria

10. ☒ Standard Review Criteria (One time use Option 1) (Taken from Attachment 5)

11. ☐ Specific Review Criteria:
☒ Source: AP-SI.1Q/Rev. 2/ICN4, Sec. 5.1.1 (One time use routine)

12. Comment Documentation:
☐ Comment Sheets ☐ Attached:
☐ Review Copy Mark-up
☒ Scientific notebook/data associated with this review as noted on Attachment 3

13. YMP-LBNL Project Manager (PM): Gudmundur S. Bodvarsson

14. Reviewer	Org./Discipline	Review Criteria	Reviewer	Org./Discipline	Review Criteria
R.F. Hedegaard	LBNL/Hydrogeologist	Technical			

COMMENTS DUE:

15. Due Date: 2/18/00

16. Originator/Review Coordinator: Lehua Pan
Print Name

REVIEW BY:

17. Randall F. Hedegaard
Print Name

18. [signature] 2MAR00
Signature Date

19. Mandatory Comments: ☐ Yes ☒ No

ORIGINATOR/REVIEW COORDINATOR (After response completed):

20. Lehua Pan Lehua Pan 3/3/00
Print Name/Signature Date

CONCURRENCE:

21. Document Draft No: NA Date: -

22. Reviewer: [signature] 2MAR00
Signature Date

23. PM: [signature] 3/3/00
Signature Date

DISPUTE RESOLUTION: (if applicable)

24. PM:
Signature Date

YMP-LBNL
COMMENT SHEET

QA: L

1. Document Title: **Routine Documentation for ExBT v1.0**			2. Page 1 of 1
3. Document No. **N/A**	4. Revision/ Change/Mod: **N/A**	5. Draft **N/A**	6. ☒ Q ☐ NQ
7. Reviewer: **Randall F. Hedegaard**			

8. NO. CODE	9. SECT./PARA./P#	10. COMMENT	11. RESPONSE	12. ACCEPT
		--NO COMMENTS— The documentation for this routine was reviewed and it was found to meet the requirements of AP-SI.1Q/Rev. 2/ICN4. The test case was checked by both hand calculation and by running the code as needed to fully check the test case. The test case fully checks the routine for the input specified and proves that the routine produces acceptable results.		

YMP-LBNL

APPLICABLE REFERENCE INFORMATION

Document No.and Title: Routine Documentation for **ExBT v1.0** per Option 1, AP-SI.1Q/Rev.2/ICN4, Sec. 5.5.1

Date of Document (or revision, draft revision number, as applicable):
NA

Pertinent sections of scientific notebook(s) or other backup documents and/or data DTN# are identified below, supporting the document which is the subject of this review. These documents/data shall be included in the scope of this review.

Document(s) Title/Data	Relevant Sections/Pages
YMP-LBNL-GSB-LP-3 Scientific Notebook	pp. 93-98

STANDARD REVIEW CRITERIA

Page 1 of 1

Routine/Macro Review Criteria, Option 1

NOTE: Where a checklist item does not apply to the software product, check "N/A".

	Yes	No	N/A	
R/M-1	✗			**The information given below is to be documented in the technical product, in which the routine/macro is used to support. Does the routine/macro include:** Name of routine/macro with version/Operating System/hardware environment
R/M-2	X			Name of commercial software used to write the routine/macros with version/Operating System/hardware used to develop it
R/M-3	X			Test Plan • Explanation whether this is a routine or macro and a description of what it does • The source code (this section shall include equations or algorithms form software setup (Labview, Excel, etc.) • Description of test(s) to be performed (be specific) • Specified range of input values to be used and why the range is valid
R/M-4	X			Test Results • Output from test (explain difference between input range used and possible input) • Description of how the testing shows that the results are correct for the specified input • List of limitations or assumptions to this test case (s) and code in general • Electronic files identified by name and location (included if necessary to perform the tests)
R/M-5	X			Supporting Information. Include background information, such as revision to a previous routine or macro or explanation of the steps performed to run the software. Include listing of all electronic files and codes used. Attach Scientific Notebook pages with appropriate information annotated.